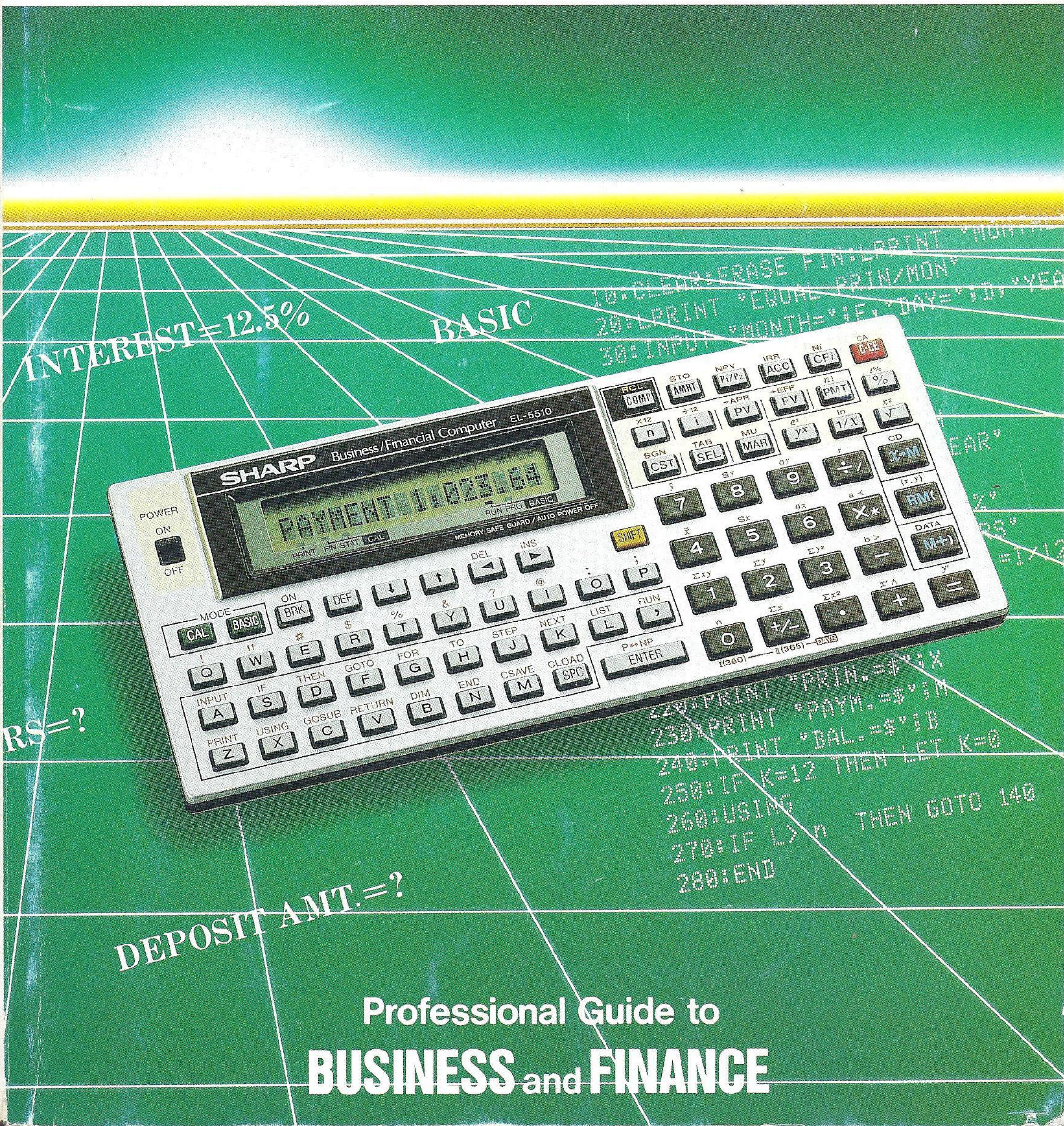
SHARP

DOLLARS AND CENTS COMPUTING

Applications for the SHARP EL-5510 Financial Computer



Professional Guide to

BUSINESS and FINANCE

NOTICE

The material contained herein is supplied without representation or warranty of any kind. Sharp Electronics Corporation therefore assumes no responsibility and shall have no liability, consequential or otherwise, of any kind arising from the use of this material.

SHARP

DOLLARS AND CENTS COMPUTING

Applications for the SHARP EL-5510 Financial Computer

Copyright © 1984
SHARP ELECTRONICS CORPORATION

TABLE OF CONTENTS

Chapter Page

1 Introduction. 1
Modes 1
Calculator Mode 2
The Normal (or usual) Math Calculations 2
Financial Calculations 12
Statistical Calculations 26

2 Intdocution to Programming 34

3 Programming Technique 38

4 Applications. 62
Business. 62
Financial 75
Statistical Tests. 82
Mathematics. 86

APPENDIX

Glossary of Terms 101

INDEX 105

NOTICE

Some answers in this text have been condensed. Your results may display more digits than those in the text. Please refer to your instruction manual for calculation ranges.

CHAPTER 1

INTRODUCTION

Whether you are engaged in Business, Financial Analysis, Banking, Marketing, or Statistics you will find that the Sharp Financial Computer (EL-5510) is an invaluable addition to the "tools of the trade". The Sharp Financial Computer is a new and very powerful computing instrument for all types of business and financial applications, many of which are presented in this booklet.

The booklet demonstrates applications that can be performed with or without programming, and provides varied examples for using the EL-5510 that will appeal to the broadest spectrum of users. Simply by pressing certain function keys, the user can directly perform calculations to solve interest, future value, amortization, cash flow and similar problems. Formidable programming capabilities permit the solution of more complicated problems. The Sharp EL-5510 instructions and applications presented in this booklet can help you become proficient in programming using the popular BASIC (Beginners All Purpose Symbolic Instruction Code) Computer Language.

This booklet was written to help you learn how to use the Sharp EL-5510, and to add to your knowledge and enjoyment of computing. Your comments and suggestions would be welcome.

MODES

We will learn first how the Sharp EL-5510 can be used as a calculator and later, as a computer, by selecting the operational modes, either the **CAL** mode, or the **BASIC** mode.

1. The **CAL** Mode: In the **CAL** mode three types of **CAL**culations can be performed, each of which is accessed by pressing the **CAL** key.

a. Normal – the computer display does not designate with "bar" symbol... usual direct math functions (multiply, add, subtract divide)

b. FINancial – the computer display designates with a "bar" which appears above the word FIN.

c. STATistical – the computer display designates with a "bar" above the word STAT.

2. The **BASIC** Mode: In this mode, there are two sub-modes, one for BASIC **PRO**gramming, another for **RUN**ning a program or calculations.

a. Selecting PRO under the BASIC Mode is required to enter BASIC PROgrams. These proceedures are covered in Chapter 3.

b. To RUN a program or to make calculations in the RUN sub-mode requires pressing the **BASIC** *key. Please refer to Chapter 3 for detailed instructions on the use of the BASIC Mode.*

Note: The SHIFT key is used to obtain the function (brown) above each key either mode.

CALCULATOR MODE

The instructions for using the Sharp EL-5510 in the **CAL**culator Mode are presented in this section within the context of practical business and financial applications, including a Balance Sheet analysis.

Part 1. The Normal (or usual) math CALculations – Addition, Subtraction, Multiplication, Division, and use of Constants.

From the accounting data of a "typical" business we perform numerous CALculations of importance to business and financial analysts, including the "Quick-Ratio". Price-To-Earning Ratios, and much more. We will call this "typical" business the BNW COMPANY, a manufacturer of boxes. First the company's financial data is presented. This data will be used throughout the manual to illustrate the many practical applications of the Sharp Financial Computer.

– BALANCE SHEET –

(In Thousands)

Assets:	
Current Assets	
Cash	*9,534*
Marketable Securities	*42,295*
Accounts Receivable	*31,217*
Inventories	*9,407*
Prepaid Expenses	*1,588*
Deferred Income Taxes	*1,102*
Total Current Assets	*95,143*
Plant and Property, at Cost:	
Land	*8,255*
Buildings	*58,467*
Equipment	*95,044*
	153,511
Less Allowance for depreciation	*62,533*
	90,978
Other Assets:	*27,546*
	221,922
Liabilities:	
Current Liabilities:	
Accounts Payable Trade	*10,113*
Accrued Wages & Commissions	*10,311*
Taxes	*2,299*
Other Payables	*8,260*
Federal & State Income Taxes	*10,959*
Advanced Payments for Product	*36,609*
Total Current Liabilities	*78,551*

Long-Term Debt	*7,055*
Other Liabilities	*4,799*
Total Liabilities	*90,404*
Stockholders Equity:	
Common Stock, authorized 18,000,000 shares, par value $1 per share; issued 16,113,484 share	*16,113*
Retained Earnings	*136,749*
	152,862
Less, Treasury Stock	*21,344*
Total Stockholders' Equity	*131,518*
	221,922

STATEMENT OF INCOME AND RETAINED EARNINGS

(in thousands)

Total Revenues:	*317,297*
Expenses:	
Production	*91,872*
Selling, Administration & General	*82,352*
Raw Materials	*61,432*
Depreciation	*9,512*
Operating Expenses	*245,168*
Operating Income	*72,129*
Other Income (deductions)	
Interest Expense	*(832)*
Interest Income	*2,727*
Other, net	*5,997*

Income before Income Taxes	*80,021*
Income Taxes	*40,988*
Net Income	*39,033*
Retaining Earning at beginning of year	*11,052*
Deduct: Dividends	*19,336*
Retained Earning at end of year	*136,749*
Per Share:	
Net Income	*2.49*
Cash Dividends	*1.20*

The market price of the company's stock on the New York Stock Exchange varied from a low of 29 to a high of 37 during the year.

Now that we have access to the company's financial condition let's use the EL-5510 to analyze these financial data using the CAL mode.

Addition

The BNW COMPNAY has cash in four bank accounts. Verify the total amount of cash as shown in the Balance Sheet:

Name of Bank	*$$ (in 1,000's)*
Chase Manhattan Bank	*$ 5,024.*
First National Bank	*$ 1,286.*
The Midland Bank Ltd	*$ 1,697.*
Hartford National Bank	*$ 1,527.*

TYPE in the number(s) then PRESS the + Key then PRESS the = Key:

5024 + 1286 + 1697 + 1527 = $ 9,534. *Answer*

Subtraction

The BNW COMPANY has most of its cash in the Chase Manhatten Bank. How much cash does it have in the other three banks?

Key in: 9534 – 5024 =

Answer: 4,510

Note: If you have placed material in the display and have not used the computer for approximately ten minutes, the EL-5510 will turn itself off. APO or Automatic Power Off automatically conserves battery life in this way.

Multiplication

The Company paid cash dividends of $1.20 per share and there were 16,113,484 shares outstanding. What was the total amount of dividends paid?

Key in: 1.2 x 16113484 =

Answer: $19,336,180.80

Note: By pressing **TAB** you can change the number of decimal places. Use **TAB** followed by the required number. Use **TAB** 2, which rounds values to 2 decimal places, for dollars and cents calculations.

Division

What proportion of income was paid out in dividends?

Key in: 1.2 ÷ 2.49 =

Answer: 0.48

Calculate the Acid-Test or "Quick" Ratio: A company's ability to pay short term obligations.

Acid-Test Ratio = (Cash + Marketable Securities + Accounts Receivable) ÷ Current Liabilities

Key in: 9534 + 42295 + 31217 = ÷ 78550 =

Answer: 1.06

Constants

What was the Price-Earnings Ratio at the high and low market prices for the year?

Key in:	*Answer:*
37 ÷ 2.49 =	14.86
29 =	11.65

The computer in the **CAL** mode is equipped with a built-in constant feature which allows repetitive calculations (calculating with the same entry without having to re-enter that number and the function key). In addition, the number entered after the + can be used as a constant. The same applies to subtraction and division. In multiplication, the number entered before the x is the multiplicand.

Reduce each of the BNW Company's current expenses by $5,000. What would be the value for raw materials if it were reduced by twice this amount?

Category:	**Key in:**	**TAB** ⊡	*Answer:*
Production	91872 – 5000	=	86,872
Selling	82352	=	77,352
Raw Material	61432	=	56,432
		=	51,432 (double deduction)

The price of gasoline is $1.20 per gallon. A car averages 30 miles per gallon. What would be the gasoline cost for trips of 400, 1000, and 250 miles?

Note: The first calculation establishes the constant.

Key in:

Constant: *Answer:*

1.2 ÷ 30 x 400 = 16

1000 = 40

2500 = 100

Trade or chain discounts are a series of discounts deducted successively from the list price. For example:

A company provides discounts of 30%, 5% and 2%. What is the final price on products having list prices of: $200, $100 and $50?

Note: Calculate the full discount by using an equivalent single discount which is the resultant of each discount multiplied by the other.

Single Discount Equivalent: $(1-r_1)\ (1-r_2)\ (1-r_3)$

$(1-0.3)\ (1-0.05)\ (1-0.02)$

$0.7 \times 0.95 \times 0.98$

Key in: *Answer:*

.7 x .95 x .98 x .6517 equivalent discount

200 = 130.34

100 = 65.17

50 = 32.585

TAB 2 32.59

The constant divisor helps create a table of index numbers. The BNW Company's revenues over a five year period were:

1980: $317,297; 1981: 289,000; 1982: 350,000: 1983: 378,000; and 1984: 419,000. The 1980 quantity becomes the constant divisor.

Key in:		**Constant:**		**Index #**	**Year**
				100	1980
289000	÷	317297	=	91	1981
350000			=	110	1982
378000			=	119	1983
419000			=	132	1984

Percent, Percent Change

The formula for computing simple interest is:

interest = principal x interest rate x time

The interest rate is usually given as a percent. The BNW Company borrowed $20,000,000 for 169 days at an annual interest rate of 9%. Calculate the interest expense.

Key in: 20000000 **x** 169 ÷ 365 **x** 9 %

Answer: $833,424.66

The BNW Company's retained earnings are $117,052 at the beginning of the year and $136,749 at the end of the year. How much is the percent increase or decrease between the two retained earnings?

Key in: 136749 – 117052 Δ%

Answer: 16.83% increase

Memory

The EL-5510's independently accessible memory is indicated by the three blue keys: **RM**, *x*→**M**, **M+**. Before starting a calculation in which memory is used, clear the memory by pressing **C.CE** and *x*→**M**.

Determine the inventory for the BNW Company.

Product:	Value/Item:	Quantity:
I	$250	11,500
II	80	20,000
III	170	20,950
IV	170	3,153
V	380	1,200

Key in: **C·CE** *x*→**M**

					Value of each Product:
I:	250	**x**	11500	**M+**	2,875,000.00
II:	80	**x**	20000	**M+**	1,600,000.00
III:	170	**x**	20950	**M+**	3,561,500.00
IV:	290	**x**	3153	**M+**	914,370.00
V:	380	**x**	1200	**M+**	456,000.00

Press **RM** to get total inventory of: $9,406,870.00

The first day of the new fiscal year, 100 of the Type V boxes were removed from inventory, calculate the new inventory.

Key in: 380 **x** 100 **+/–** **M+**

Answer: –38,000.00

Press **RM** New inventory is $9,368,870.00

Note: The keys **+/–** **M+** are used to subtract from Memory (M–).

The EL-5510 has special business function keys which are described in the following sections.

Mark-ups

The retail price of an item is equal to its wholesale cost plus the markup added by the retailer. One of the BNW Company's products, the black and white box, has a cost of $68. What is the retail price if the markup on cost 60%?

Key in: 68 **CST** 60 **MU** **COMP** **SEL**

Answer: $108.80

If the retail is $100 what is the percent of markup on cost and the gross margin?

On cost, Key in: **TAB** 0 100 **SEL** 68 **CST** **COMP** **MU**

Answer: 47% markup

Gross margin, Key in: 100 **SEL** 68 **CST** **COMP** **MAR**

Answer: 32%

Find the cost on a $100 retail item with a markup of 40% on retail and gross margin of 40%.

On Cost, Key in: **TAB** 2 110 **SEL** 40 **MU** **COMP** **CST**

Answer: $78.57

Gross Margin, Key in: 40 **MAR** **COMP** **CST**

Answer: $66.00

Calendar

There are two methods used for calculating the number of days between two dates. The first is based on a standard 30-day month, 360-day year. This approach is usually used for ordinary simple interest, simple real estate loans, installment loans, and periodic equipment loans.

The second method is based on a 365-day year and an exact number of days. This method is used to determine accurate interest and is the most commonly used. The EL-5510 includes an extra day for Februarys occuring in a leap year

a. *Calculate the number of days from February 1, 1984, to March 1, 1984 using each method:*

Key in:	month/day/year			TAB ⊡		Answer:
	2.	01	1984	I (360)	3.011984 =	30
	2.	01	1984	II (365)	3.011984 =	29

b. Calculate February 1, 1985 to March 1, 1985

						Answer:
	2.	01	1985	I (360)	3.011985 =	30
	2.	01	1985	II (365)	3.011985 =	28

c. Calculate February 1, 1980 to February 1, 1985 using each method.

						Answer:
	2.	01	1980	I (360)	2.011985 =	1800
	2.	01	1980	II (365)	2.011985 =	1827

Part 2: Financial Calculations

The special business programs built into the EL-5510 are only useable in the FIN mode. Before beginning any calculation, clear with **CA**.

Financial calculations are often long and complicated any may occasionally take up to a minute to complete. If you want to interrupt a long calculation, press the C·CE button.

The EL-5510 offers three basic types of financial calculations:

Interest
Amortization
Discounted cash flow analysis

INTEREST

Interest calculations are necessary to determine the time value of money. To compute interest calculations four keys are required:

n = Number of compounding periods
i = Interest value per compounding period

PV = Present Value of money
FV = Future Value of money

The basic equation is: $FV = PV(1 + i)^n$

AMORTIZATION

Amortization refers to the type of installment loan known as direct reduction. The periodic (usually monthly) payments are always the same. However, the greatest proportion of the interest is paid on the first payment and smallest on the last. The principal returned follows the opposite pattern.

Illustration

To the keys described above we add:

PMT = PayMenT
AMRT = gives the principal portion and interest portion for a particular payment period – remaining principal.

P1/P2 and ACC = gives the total of principal and interest for a specified time period.

The basic equation is $PV = PMT \dfrac{1-(1+i)^{-n}}{i}$

DISCOUNTED CASH FLOW

An investment is worthwhile only if it earns a return considered acceptable to the investor. A method used by financial analysts to determine the present values of a series of cash flows is called Net Present Value (NPV). For an interest rate specified (return on investment), a positive NPV indicates a desireable investment.

How much will an investment earn on average over the life of a particular project? Since any business venture requires large amounts of capital at start-up, and sales of land, equipment or products are realized over time or at a project's termination, a technique called "DISCOUNTED CASH FLOW" is used to determine the average rate of compound interest earned by a project on money invested over the life of the project. The DCF discounts the annual cash flow (sum of net profit and depreciation) to properly reflect the time value of money invested on a continuous compound interest basis.

i = anticipated rate of return
CFi = to enter the cash flow amount for each period
Ni = to enter a series of identical cash flows
NPV = calculates the net present value
IRR = calculates the internal rate of return or yield on an investment.

NOTE: BEGIN

Whenever payments are made, it is necessary to specify whether they are made at the beginning or the end of the payment period. The payment key on the EL-5510 is marked BGN. If BGN appears in the display, payments will be calculated as if the payment occured at the beginning of the month. Press BGN and the display will be cleared of BGN; payments will be calculated as if they were made at the end of the period.

INTEREST

The Treasurer of the BNW Company, Ed Green, has $1,000,000 to invest for th period from June 30, 1984 to December 1, 1984. The Liquid Money Fund and the Fidelity Fund both pay an annual interest rate of 11% compounded daily. Fidelity uses actual days. Liquid uses standard 30-day months. Which Fund will provide Mr. Green with the greatest return?

Key in: CAL (change to FIN mode)

MODE FIN:

Key in: CA TAB 2
1000000 +/– PV

NOTE:

The convention in future value calculations is to call the present value a negative quantity. Payments to the Bank or to a supplier will appear as negative amounts in the display.

a. Liquid Money Fund

Key in:

6.301984 **I (360)** 12.011984 =

151 **n**

11 ÷ 360 = **i**

COMP FV *Answer:* $1,047,212.47

x→**M**

b. Fidelity Fund

Key in:

6.301984 **II (365)** 12.011984 =

154 **n** 11 ÷ 365 = **i**

COMP FV *Answer:* 1047497.48 – **RM** =

Answer: 285.01

The Fidelity Fund would provide a return of $285.01 more than the Liquid Money Fund.

The effective rate of interest will usually be different than nominal rate of interest (APR = Annual Percentage Rate). Compounding more often than once a year will result in an effective rate higher than the nominal rate.

Calculate the effective rate for an APR of 8% compounded quarterly, monthly and daily.

Key in:	*Answer:*
TAB 3	
4 **EFF** 8 =	8.243
12 **EFF** 8 =	8.300
365 **EFF** 8 =	8.328

An effect interest rate of 7.25% is quoted on an account compounded monthly. What is the APR?

12 **APR** 7.25 = 7.020%

Interest Applications:

REGULAR SAVINGS

Mr. Black has decided to open a savings account. He will deposit $200 initially and $50/month. The bank pays 5.25% compounded monthly. What will be the value of the account in 3 years.

Key in: **CA** **TAB** 2
200 **+/–** **PV**
50 **+/–** **PMT**
5.25 **÷12** **i**
3 **x12** **n**
COMP **FV**

Answer: 2,178.94

ACCOUNT DEPLETION

A trust fund is set up by the BNW Company to help pay the tuition of a deserving college student with financial needs. A contribution of $10,000 is made from which $300/month will be withdrawn. If the rate of interest is 5.5%, how long will the account provide the monthly payment?

Key in: **CA** **TAB** 2
10000 **+/–** **PV**
5.5 **÷12** **i**
300 **PMT**
COMP **n**

Answer: 36.26 months or 3 years

DEPLETION WITH LIMIT

If it is required that the account stop making payments when it reaches $2,000, how many months will the $300/month be available?

Key in: 2,000 **FV**
COMP n
Answer: 29.47 months

SET MONTHLY PAYMENTS

In order to meet current expenses, the BNW Company took a loan from the bank of $100,000 at 12% compounded monthly. It was agreed that payments would be made back to the bank at $20,000/month. How many months would it take to pay back the loan?

Key in: **CA TAB** 2
100000 **PV**
12 **÷12 i**
20000 **+/– PMT**
COMP n
Answer: 5.15 months

ADD-ON INTEREST

Another loan is taken for one year at $50,000; interest at 10% is added on. What is the actual rate of interest?

Key in: **CA TAB** 2
50000 **PV**
+ 10 %
÷12 +/– PMT
12 **n**

COMP **i**

x12

Answer: 17.97

BONDS

The value of a bond is calculated by taking the Present Value of the coupon payments plus the Present Value of the principal or its redemption value at a given interest rate. To properly calculate the yield, discount, interest and price of a bond, one must consider the formula that applies to the bond, the accrued interest or discount, concessions, and if it is callable, the call date. For calculations of shorter term amounts the day count basis is needed and the number of days between pertinent dates.

PRESENT VALUE OF A BOND

The BNW Company has $3,000,000 in long term debt that consists of 4% bonds, with $1,000 face value and 11 years to maturity. The payment of $20 is semi-annual. An investor wishes to buy the bonds to yield 6%. What would be the present value of the bonds?

Key in: **CA** **TAB** 2

6 **i**

11 **n**

1000 **FV**

COMP **PV**

Answer: PRESENT VALUE OF BOND: $526.79 *x*→**M**

Key in: 20 **PMT**

6 ÷2 = **i**

11 x 2 = **n**

0 **FV**

COMP **PV**

Answer: PRESENT VALUE OF COUPONS: $318.74

TOTAL PRESENT VALUE FOR BOND:

+ **RM** = 845.53

SERIAL ISSUE

What is the average value of a serial issue of $400,000 in 8% bonds maturing 40,000 each half year for 5 years if bought to yield 7%?

The general formula for solving this problem is called "Makeham's Formula".

$$\textbf{DESIRED VALUE OF LOAN} = K + \frac{j}{i}(C - K)$$

Where C = capital repayable

j = the nominal rate/period

i = yield/period

K = present worth of C based on i

in this case: $c = 40000$ $j = 4\%$ $i = 3.5\%$

CA TAB 2

3.5 **i**

10 **n**

40000 **+/–** **PMT**

COMP **PV**

Answer: $332,664.21

Final Answer, Key in:

400000 – 332,664.21 = *x* →**M**

4 ÷ 3.5 x **RM** = + 332,664.21 =

Answer: $409,619.40

SINKING FUND

Sinking funds are usually established by a corporation to pay back the principal of a bond issue when it becomes due.

What payment is necessary to a sinking fund to retire a $10,000,000 loan in 20 years, with semi-annual payments and annual interest at 8%?

Key in: **CA TAB** 2
10000000 **FV**
8 ÷ 2 = **i**
20 x 2 = **n**
COMP PMT

Answer: $105,234.89 (semi-annual payment)

What is the time required to extinguish a $2,000,000 loan with annual fixed payments of $200,000 and 10% annual interest?

Key in: **CA TAB** 2
10 **i**
2000000 **FV**
200000 **+/– PMT**
COMP n

Answer: 7.27 years

AMORTIZATION

The BNW Company is planning to relocate its corporate office this year to Greenwich, Conn. The president of the company, John Brown, will relocate to the southwestern Connecticut area. Mr. Brown intends to purchase a small home for $140,000. He will pay $70,000 down and incur a mortgage obligation of $70,000 at 12.5% interest for a term of 30 years. What will Mr. Brown's monthly payment be?

Key in: CA TAB 2
70000 PV
12.5 ÷ 12 i
30 x 12 n
COMP PMT

Answer: $747.08/ Month

What would be the monthly payment for a 25-year mortgage?

25 x 12 n
COMP PMT

Answer: $763.25 / Month

Using the 25-year mortgage:

a. Calculate the principal paid, interest paid, and remaining principal after the first payment:

Key in:		**Upper Display**
1 AMRT	$34.08	PRN
AMRT	$729.17	INT
AMRT	$69,965.92	BAL

b. At the 150th payment:

Key in:		
150 AMRT	$159.62	PRN
AMRT	$603.63	INT
AMRT	$57,788.56	BAL

c. What would be the amount of principal and interest paid during the second year of the mortgage? (the 13th through the 24th payment period)

Key in:

13 **P1/P2** 24 **P1/P2**

ACC *Answer:* $490.61 **PRN**

ACC *Answer:* $8,668.37 **INT**

d. What would be the principal and interest paid during the entire mortgage?

Key in: 1 **P1/P2** 300 **P1/P2**

ACC *Answer:* $70,000 **PRN**

ACC *Answer:* $158,974.37 **INT**

NOTE: The values in an interest problem are automatically stored in the store memories. They can be recalled as follows:

Key Equivalent	Name
RCL i	interest
RCL n	number of periods
RCL PV	present value
RCL FV	future value
RCL PMT	payment

POINTS

Many banks in order to increase their return on mortgages charge points. 1 point is equal to 1% of the mortgage principal. If the bank had charged Mr. Green in the example previously described, 2 points, what would have been the actual rate of interest?

Key in: 25 **x 12 n**

12.5 **÷12 i**

70000 **PV**

COMP PMT $763.25

RCL PV **x** 2 **%** 1400 ***x*→M**

RCL PV – RM = 68,600 **PV**

COMP i 1.07

x12

Answer: 12.80%

FEES

In addition to the points, there was a $150 fee charged to the mortgage. What would have been the actual rate of interest?

Key in: **RCL PV –** 150 **=** 68,450 **PV**

COMP i 1.07

x12

Answer: 12.83%

DISCOUNTS

Mr. Green holds the mortgage on a house he sold 2 years ago. He wishes to sell the mortgage, but present buyers insist on a yield of 12%. The current mortage is at 8.5%, $46,000 and 26 years to go. What price would he get for this mortgage?

Key in: **CA TAB** 2

8.5 **÷12 i**

26 **x12 n**

46000 **PV**

COMP PMT $366.33

Change i: 12 **÷12 i**

COMP PV

Answer: $34,990.59

Mr. Brown is offered a mortgage with a remaining principal of $30,000, original annual yield of 7%, 10 years remaining, at $25,000. What would the yield be assuming that monthly payments remain the same?

Key in: 30,000 **PV**

7 **÷12 i**

10 **x12 n**

COMP PMT $348.33

Change present value: 25,000 **PV**

COMP i 0.94

x12

Answer: 11.28%

BALLOON PAYMENTS

Commonly used by corporations to establish second mortgages on properties that they hold. The BNW Corporation is taking a mortgage on a new building for $150,000. The mortgage is for 14 years and a balloon payment of $30,000 will be made as the last payment. The interest rate is 9.25%. What is the monthly payment?

1. CALCULATE THE PRESENT VALUE OF $30,000

Key in: **CA TAB** 2

30,000 **FV**

9.25 **i**

14 **n**

COMP PV

Answer: $8,694.03

2. PRESENT VALUE

150,000 − 8,694.03 = 141,305.97

141,305.97 **PV**

0 **FV**

3. THE NUMBER OF PAYMENTS IN 14 YEARS LESS ONE MONTH

13 **x12** **+** 11 **=** **n** 167

4. 9.25 **÷12** **i** .77

COMP **PMT**

Answer: $1,507.35/month

DISCOUNTED CASH FLOW

The BNW Company is considering buying a warehouse, and then selling it after a four year period. The building will cost $100,000 and Mr. Green plans to sell it for $120,000, 4 years later. The equivalent savings in rent the first three years are anticipated to be $7,000, $8,500, and $9,000. Will this investment meet their objective of a return of 10% a year?

MODE: **FIN**

Key in: **CA** **TAB** 2

10 **i**

100000 **+/–** **CFi** (If the analysis includes an investment, the initial cash flow must be entered as negative. If no investment is involved, the initial cash flow entry should be 0.)

7000 **CFi**

8500 **CFi**

9000 **CFi**

120000 **CFi**

NPV (net present value)

Answer: $2,111.88 – The answer for NPV is positive, therefore the return will be greater than 10%.

IRR (internal rate of return)

Answer: 10.64 – Actual yield for this example.

Note that the maximum number of allowable cash flows (a cash flow may or may not include multiple cash flows using Ni function) is 20. The Ni function allows entry of up to 99 consecutive identical cash flows (See Applications, Chapter 4 for example).

To recall cash flows 0–9, use **RCL** followed by a number 0–9. To recall cash flows 10–19, use **RCL** followed by ⊡ and then the number.

To clear a value use **STO**. Change the value 9,000 to 6,000 in the previous example.

Key in: 6000 **STO** 3
RCL 3

Answer: 6,000

Part 3: Statistical Calculations

The EL-5510 has single-variable and two-variable statistical functions, and Linear regression.

To clear previous statistical inputs and calculations press **CA.**

Single-variable Statistics

$\bar{x}$ represents the mean, or arithmetic average. The formula for $\bar{x}$ is:

$$\bar{x} = \frac{x_1 + x_2 + x_3 + \cdots x_n}{n}$$

n is the number of samples. If the data base is small, it could be the total number. x_1 is the first data point, x_2 is the second up to x_n, which is the nth or last data point. The symbol Σ (sigma) means "sum of" and when Σx is pressed, the total of all the data is retrieved.

The statistical techniques available in the calculator mode are based on the concept of central tendency. A large enough data base will result in a frequency of data about the mean with values concentrated close to the mean, resulting in the familiar normal (or bell-shaped) curve illustrated in Figure 1.21.

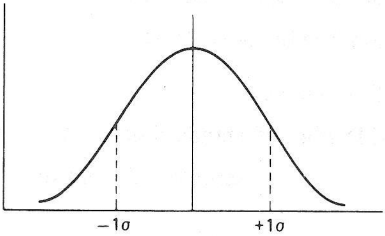


Figure 1.21 Normal Distribution

Standard deviation is the measure of deviance of data from the mean. A large deviance implies a group of data with less cohesion or greater deviance than the case where the deviance is small. One standard deviation (plus and minus) includes 68.27% of the area under the normal curve. Two standard deviations will include 95.45% of the total and 3 includes 99.73%. $\mathbf{S}_x$ is the standard deviation of the sample and σ_x the standard deviation of the population. Logically, the standard deviation of the whole population will always be less than the standard deviation of a sample. The basic formulas for standard deviation are:

$$\sigma = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n}} \qquad s = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n-1}}$$

The general procedure for obtaining the average and standard deviation for a set of data is:

1. Place the calculator in **STAT** using the **CAL** key.
2. Place the data into the calculator using the **DATA** key.
3. Press $\bar{x}$ for the average and $\mathbf{S}_x$ for standard deviation.

The price of an eight ounce jar of instant coffee was recorded each month of the year. Calculate the average price and the standard deviation from the prices:

$3,65, $4.20, $4.20, $4.85, $3.90, $3.90, $2.95, $2.80, $3.90, $3.50, $3.50, $3.20.

Mode: **STAT**

Key in: 3.65 **DATA** 4.2 **DATA** 4.2 **DATA** 4.85 **DATA** 3.9 **DATA**···
(Continue to place in above data)

Answer: $\bar{x}$ = 3.7125 = average
σx = 0.551182441 = standard deviation for the population since the sample is the entire population.

Under the assumption that the sample is large enough for the normal curve to apply, ranges for the prices can be obtained by taking the mean and then adding or subtracting the standard deviation.

Key in: **TAB** 2	*Answer:*
$\bar{x} - \sigma x =$	3.16
$\bar{x} + \sigma x =$	4.26

Approximately 68% of the prices during the year fell between $3.16 and $4.26. How many standard deviations will encompass all of the sample prices? Subtract the lowest data point from the average and divide by the standard deviation for the negative number. For the positive number, take the highest data point, subtract the average and divide by the standard deviation. If the value obtained is fractional, always round up. For example, if it is 1.2, the number of standard deviations is 2.

Key in:	*Answer:*
$\bar{x} - 2.8 = \div \sigma x =$	1.66 (or –2 standard deviations)
$4.85 - \bar{x} = \div \sigma x =$	2.06 (+3 standard deviations)

Data input with "weighting" is possible. Weighting is particularly valuable where a large number of samples have been collected of the same value. The value inputted is being weighted by entry more than once in a single operation.

Value	Number of Occurences
6	10
7	48
8	49
9	32
10	17

The number of times (N) the data occurs is entered as x N. After pressing **DATA** each time, the total number of data points entered will be displayed.

MODE: **STAT**

Key in: **CA**

6 x 10 **DATA** 7 x 48 **DATA** 8 x 49 **DATA**
9 x 32 **DATA** 10 x 17 **DATA**

n is 156	(use **TAB** 3)
$\bar{x}$ is 7.987	mean
S_x is 1.101	standard deviation for the sample because the sample is a small part of a large population.

Standard deviation in this example means 68.27% of the values fall between:

Key in:		*Answer:*
	$\bar{x}$ + S_x =	9.088 and
	$\bar{x}$ – S_x =	6.886
95.4% between	2 x S_x + $\bar{x}$ =	10.190 and
	2 x S_x +/– + $\bar{x}$ =	5.785
99.73% between	3 x S_x + $\bar{x}$ =	11.291 and
	3 x S_x +/– + $\bar{x}$ =	4.684

CD means "correct data." If a data entry was in error or it will be changed, re-enter it and then press **CD**. The **CD** key may be used as **DATA** entries are being made or directly after a statistical function. The number of samples will be shown, less one. Now proceed as before to enter new data. If weighting has been used, the entire original entry should be keyed in; for example, if 10 **x** 17 has been keyed in and **CD** is pressed, all seventeen of these entries will be eliminated.

Two Variable Statistics:

Two-variable statistics makes possible the development of a relationship (correlation) between two sets of data. Each pair of data has an x and y value.

From these sets of data a line of regression can be established. The relationship of the two sets of data by use of the straight line method is called "Linear Regression". The equation of the straight line is $y = a + bx$, where a is the point at which the line crosses the Y-axis and b is the slope of the line.

The correlation coefficient is called "r".

$$r = \frac{n\Sigma(xy) - \Sigma(x)\,\Sigma(y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2]\;[n\,\Sigma y^2 - (\Sigma y)^2]}}$$

A perfect correlation between two values in linear regression results in an r equal to 1 (−1 is a perfect negative correlation); in other words, by knowing the value of one variable you can predict with 100% accuracy the value of the other variable. The further the value of r is from ±1, the less reliable will your predictions be. The following table can be used as a set of definitions of the values of the correlation coefficient:

	Value of *r*	**Call it**
Positive Correlation	+0.80 to 1.00	Extra High
	+0.60 to 0.80	High
	+0.40 to 0.60	Moderate
	+0.20 to 0.40	Low
Negative Correlation	−0.20 to +0.20	Nil
	−0.20 to −0.40	Low
	−0.40 to −0.60	Moderate
	−0.60 to −0.80	High
	−0.80 to −1.00	Extra High

It weight a good predictor of longevity among men 65 years of age? In 1950, 10 men, each six feet tall, were selected for an experiment to determine if their weight effected their life span.

Sample	1	2	3	4	5	6	7	8	9	10
Age at death	72	67	69	85	91	68	77	74	70	82
Weight at age 65	185	226	200	169	170	195	175	174	198	172

Key in: STAT Mode TAB 3

72 (*x*, *y*) 185 DATA
67 (*x*, *y*) 226 DATA
etc.

r −0.793

r indicates a relatively high negative correlation. Higher weight indicates a shorter life span. To graph the regression line, *a* and *b* are used.

Key in:	*Answer:*
a	321.929 (y-axis)
b	−1.795 (slope)

Predict the age of death of a 6 foot man weighing 190 pounds in 1950.

Key in:	*Answer:*
190 x'	73.495

To reach age 90, what should a man's weight be in 1960?

Key in:	*Answer:*
90 y'	160.371 pounds

To reach age 150, what should a man's weight be? Obviously, he answer will make no sense, indicating the danger of carrying a straight line extrapolation too far.

A chemist measures the pressure in a gas filled container as it is heated. From the ideal gas equations, he knows that as temperature increases and volume is held constant, pressure will also increase. Calculate the average increase in pressure (Y) per degree Celsius increase in temperature (X).

The slope of the regression line will be the increase.

X = Temperature °C	23	28	35	37	58	78	100
Y = Pressure Kg/cm^2	75	79	81	81	87	92	98

Key in: **(STAT MODE) CA** 23 (x, y) 75 **DATA**
28 (x, y) 79 **DATA**
: :
100 (x, y) 98 **DATA**

n = 7 data entries
slope = **b** = 0.278869494 $Kg/cm^2/°C$
coefficient = **r** = 0.993576328 indicates excellent data fit to least squares line.

Exponential Regression

Data points having a linear relationship can be fitted to a regression line. Many situations arise where the data can be fitted more closely to the exponential curve $y = ae^{bx}$.

Exponential data is entered by taking the natural log of the y value and the equation becomes:

$$\ln y = \ln a + bx$$

Estimated results are obtained by raising the result exponentially. The value in register a will be $\ln a$.

Determine the actual equation, coefficient of correlation, and estimated y values for $x = 9.2$ and -2.6, for the following values:.

x	0.5	1.6	3.6	7.9	8.7
y	8	13.2	52.9	1008	2201

Key in: **(STAT MODE) CA**

.5 (x, y) 8 **ln** **DATA**
1.6 (x, y) 13.2 **ln** **DATA**
3.6 (x, y) 52.9 **ln** **DATA**
7.9 (x, y) 1008 **ln** **DATA**
8.7 (x, y) 2201 **ln** **DATA**

Key in:	*Answer:*
r	0.998657732
a	1.582752611
b	0.687295479
a e^x	4.868338026

Obtain estimated values of y:

Key in:	**Estimated** y:
9.2 y' e^x	2713.164701
2.6 +/– y' e^x	0.815286548

The calculator is automatically performing the following:

$$\ln y = \ln a + bx = 1.5827 + 0.687x$$

$$y = 4.868\,e^{0.687x}$$

CHAPTER 2

INTRODUCTION TO PROGRAMMING

Programming may be thought of as a series of steps. These are: analysis of the problem, preparing an algorithm, flowcharting, writing of the program, testing and editing, and documentation. Each of these steps is performed by the programmer. These steps apply to all programmable machines from calculators to complex computers.

STEP 1: DEFINING THE PROBLEM

A well defined problem has two elements, first, the clear and precise statement of the situation, and second, a recognizable end result.

An example of a well defined problem is:
What is the area of a circle, if the radius is 10 inches?

STEP 2: ALGORITHMS

Once a problem has been clearly defined, a method of solution is determined. The desired result is obtained by establishing a series of events called an algorithm.

The computer can only perform a single operation at a time. We are forced to think in sequential terms because of the nature of the computer. From the computer's point of view, the algorithm is the order in which it must perform a set of operations.

The basic formula for the problem in Step 1 is: Area of a circle = Radius squared multiplied by constant Pi.
The alorithm for this problem is:

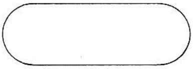
A. Take the value of the radius
B. Square this value

C. Multiply the result of “B” by Pi

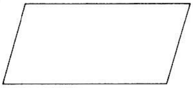
D. Display the result

STEP 3: FLOWCHARTING

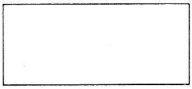
The logic of the algorithm can be visually presented and checked by the use of a flowchart. Each type of operation in the algorithm has a distinct shape. The basic shapes that apply to the handheld computer are:



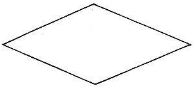
A. Oval — Signifies the beginning and ending of a program



B. Parallelogram — Input and output of data

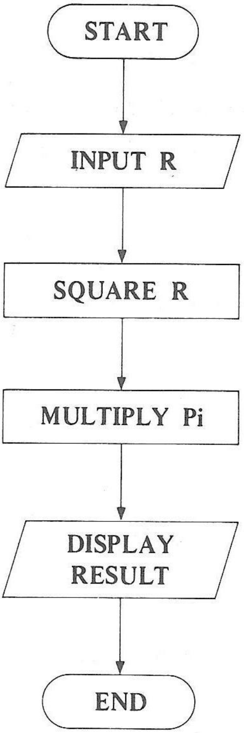


C. Rectangle — An operation (multiplying, division, will be carried out)



D. Rhombus — A decision will be made

FLOWCHARTING THE PROBLEM OF THE AREA OF A CIRCLE:



1. Take the value of the radius
2. Square this value
3. Multiply by Pi
4. Display the result

STEP 4: PROGRAMMING

The next step is to write the actual program. The EL-5510 uses a computer language called **BASIC**. **BASIC** is the most popular of the computer languages and is available on most pocket computers and micro computers. It is simple and relatively easy to learn. Detailed discussion of the EL-5510's **BASIC** is the subject of the next chapter.

STEP 5: TESTING AND EDITING

Use data with a known answer to the problem to check a program. The flowchart provides a means of determining where errors may have occurred. Errors in programming technique will result in an error code. Each error message is described in detail in Chapter 3.

STEP 6: DOCUMENTATION

A formal method of keeping program records should be developed. Many problems will have more than one variable. The order of entry of each item in a set of data must be specified if the program does not ask for each data point by name or symbol.

The program's documentation consists of the flowchart, the program, commentaries for further reference, and user instructions.

CHAPTER 3

PROGRAMMING TECHNIQUE

Starting to Program — Modes, Line Numbers

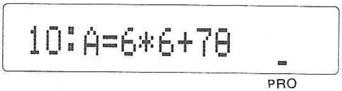
The Sharp EL-5510 uses a computer language called BASIC which stands for **Beginners All-Purpose Symbolic Instruction Code.** An instruction line in a BASIC program is called a line of code. Each line of code begins with an identifying number. Usually these numbers are in multiples of 10. Sufficient room is thereby left to insert additional lines if this should become necessary. The line number tells the computer the order by which the instructions are to be executed.

ENTER, END, NEW

Press the **BASIC** key until a bar appears over **PRO**. Type the word **NEW** (this erases any program you may have stored in memory) and press the **ENTER** key. Now write the following program which computes a simple formula and assigns a value to the variable A.

Type in: 10A = 6 * 6 + 78

Press **ENTER**



```
10:A=6*6+78
```

The **ENTER** key is used in the **PRO**gramming mode to enter each line of code into the program. After **ENTER** is pressed a colon appears between the line number and the expression. The colon should never be pressed after a line number as it will always be inserted automatically by the computer.

The computer needs to know that a program has ended. To end the above program add the line:

20 **END ENTER**

The **END** command tells the computer to stop working.

LET

The expression for A above is a **LET** statement. **LET** statements apply to both numeric and alphanumeric variables. **LET** is omitted except in a **THEN** statement as illustrated later. A series of **LET** statements on the same line are separated by colons.

RUN, BRK, STOP, CONT

The value of A is calculated by changing to the **RUN** mode. Press the **BASIC** key until a bar appears over **RUN**. Now key in **RUN** (either R-U-N or **SHIFT** [,]) and press **ENTER**. To display the value of A, press A and **ENTER**. The answer is 114.

BRK is the Break key. It is not programmable. Use it to stop a program in execution. The display will show:

"Break in (line number)"; indicating where the program has stopped.

STOP makes a program stop executing each time this command is encountered. **CONT** and **ENTER** is used to restart the program execution.

PRINT, PAUSE, WAIT

The most common means of outputting data on the display is the command **PRINT**. **PRINT** is always followed by an expression. Without an expression a blank line will result.

The **PRINT** expression may be numeric or alphanumeric (a string). Quotation marks must be used with a string. Although an expression can be longer than 16 characters, only 16 will appear on the display.

A numeric expression may be a formula. A series of formulas can be included in a statement if separated by a comma or semicolons. The display automatically divides into two sections of 8 characters each when a comma is used. Therefore only two items can be displayed simultaneously with a comma separation. The semicolon must be used if more than two items are displayed at the same time. With semicolons, ansswers will appear on the display without separation. Switch back to **PRO** mode, and key in:

```
10  X = 5 :  Y = 2              ENTER
20  PRINT  2 * X,  5 * Y        ENTER
30  END                         ENTER
```

Change to **RUN** mode; press **RUN** and **ENTER**.

Answer:

```
        10.     _  10.
                RUN
```

Change Line 20 (**PRO** mode) and rerun (**RUN** mode)

```
20  PRINT  2 * X/4;  5 * Y;  X/Y;  -X * Y
```

Answer:

```
2.510.2.5-10._
                RUN
```

Note that the answers are bunded together. Go back to **PRO** mode and change line 20 as follows:

```
20  PRINT  2 * X/4; "SPC"; 5 * Y; "SPC"; X/Y; "SPC"; -X * Y
```

Now change back to **RUN** mode and rerun.

The formula for the monthly equalized payment of a loan mortgage is:

$$\text{Payment} = PV * i/(1 - (1 + i)^{-n})$$

where: PV = Present Value (amount of the loan)
i = Annual Interest Rate (decimal)
n = Number of Payments

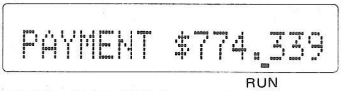
To change to a monthly basis i is divided by 12 and n multiplied by 12.

What is the monthly payment on a $70,000 mortgage at 13% for 30 years? Using the BASIC commands we have learned so far, the program for this problem is:

```
10  PRINT "MORTGAGE PAYMENT"
20  V = 70000:  I = .13:  N = 30
30  P= V * (I/12)/(1 - (1 + (I/12))^(-N * 12))
```

```
40 PRINT "PAYMENT $"; P
50 END
```

Change to the **RUN** mode; press **RUN** and **ENTER**. The display shows MORTGAGE PAYMENT. Press **ENTER** a second time to get the second **PRINT** statement which is the answer:



```
PAYMENT $774.339
```

PAUSE may be used instead of **PRINT. PAUSE** is a valuable command where it is desirable to make a statement and then have the program continue without having to press **ENTER**. Substitute **PAUSE** in Line 10 above and rerun the program.

The **WAIT** statement with a number permits the display of the **PRINT** statement for a specified period of time. 640 is the equivalent of 10 seconds. Insert **WAIT** 320: on Line 40 before **PRINT** and rerun the program.

LIST, Editing

LIST in the programming mode will produce the first line of your program. The same result is obtained by pressing the ↓ key, which will also take you through your program one line at a time. If you start with the ↑ key, the last line appears first. The advantage of the **LIST** command is it may be used with a line number to quickly access a particular statement in a large program.

The display will only show the first 16 characters of the line. To access the rest of the line the cursor keys are used. The cursor keys (◀ and ▶) allow editing.

To duplicate Line 40 as Line 45 in the previous program, bring Line 40 to the display. Press the left directional cursor key until it is over the 0; press 5 and **ENTER**.

To change the values on Line 20 of the program to $7,000, .113, and 29, go to Line 20 of the program. Press the right cursor key until the cursor is over a 0 in 70,000; **DEL** (delete). Continue to move the cursor to over the 3 in .13; **INS** (insert). To insert always move the cursor over the character that will follow the inserted character. A flashing top and bottom bracket appears; press 1. This value now is .113. Move the cursor over the 3 in 30; press 29 and **ENTER**. Rerun the program. The answer is $68.54356158.

INPUT

In the mortgage payment example, the method used to place the data into the formula was limited. Variables may be used instead with the **INPUT** command. Change Line 20 in **PRO** mode and the program becomes:

```
10 PAUSE "MORTGAGE PAYMENT"
20 INPUT V, I, N
30 P = V * (I/12)/(1 - (1 + (I/12))^(-N * 12))
40 PRINT "PAYMENT $"; P
50 END
```

Now in **RUN** mode, as the program is run, the display shows a ?. Place the value (70000) in and press **ENTER**. Continue for the next two values. Repeat changing only the interest rate to 17%. The new answer is: $997.9727411.

Rather than have only question marks appear it is safer to name the variables. To achieve this the format of Line 20 is changed to (in **PRO** mode):

```
20 INPUT "V="; V, "I="; I, "N="; N
```

Run the program again — note the placement of semicolons and commas in Line 20.

The **INPUT** command also applies to strings. In the example include the owner's first name. Add:

```
 5 INPUT "NAME ?"; A$
15 PAUSE A$; "S HOUSE"
```

Rerun the program using your first name as the first input.

IF . . . THEN, GOTO, BEEP

The command **IF . . . THEN** allows decisions to be incorporated into a program. **IF** is followed by an equation or inequality and **THEN** by a statement number.

Good programming technique dictates that the number of the statement following **THEN** should not be number of the next statement in the program. In an **IF . . . THEN** statement the computer will test the condition stated and:

1. If true, will branch to the statement after **THEN**
2. If false, ignores the **THEN** clause and continues with the next line of the program.

The conditionals useful after **IF** in BASIC are:

Symbol	Meaning
=	Equals
<	Less than
>	Greater than
<>	Less or greater than
<=	Less than or equal to
>=	More than or equal to

Returning to the mortgage payment problem, to repeat the program without having to press **RUN** each time, add:

```
50  INPUT “REDO (Y/N) ?”; A$
60  IF A$=“Y” THEN 20
70  END
```

The entire program now is:

```
10  PAUSE “MORTGAGE PAYMENT”
20  INPUT “V=”; V, “I=”; I, “N=”; N
30  P = V/((1 – ((1 + I/12)^(–12 * N)))/(I/12))
40  WAIT 320: PRINT “PAYMENT $”; P
50  INPUT “REDO (Y/N) ?”; A$
60  IF A$=“Y” THEN 20
70  END
```

The calculations for the mortgage payment program can be accomplished by using the financial functions built into the EL-5510. V is the key PV; I divided by

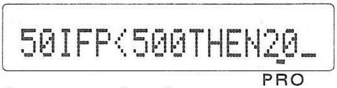
12 is i; and N times 12 is n. "Divided by 12" and "Times 12" are single key entries. Change lines as follows:

```
30  i = I /12:  n = N *12:  PV = –V
40  WAIT 320: PRINT "PAYMENT $"; COMP PMT
```

Calculate the monthly payment for mortgages of $40,000; $50,000; and $60,000 with a 15% interest rate and 30 year term. Press **RUN**.

V = 40000 I = 15 N = 30 Answer: $505.777
at REDO (Y/N)? input Y
V = 50000 I = no entry Answer: $632.222
REDO Y
V = 60000 I = no entry Answer: $758.666
REDO N

If the buyer wishes to keep the monthly payment below $500, the program can be changed to read:



```
50IFP<500THEN20_
```

PRO

Test 3 mortgages to see if they meet the above criteria:

a. $40,000, 12%, 25 years
b. $30,000, 15%, 30 years
c. $50,000, 15%, 30 years

Mortgage "c" resulted in a payment in excess of $500 per month and therefore the program ended.

GOTO cycles a program to a previous or forthcoming statement line. In the mortgage example to continuously run the program, the lines 50 and 60 are replaced by:

```
50  GOTO 20
```

The **GOTO** statement can be used instead of **THEN**.

LOOPING

To limit the number of times a calculation can be performed or to number an operation a counter is used (looping). The most common form of counter is the **LET** statement:

```
I = I + 1
```

Each time this statement is passed in the program, the computer adds 1 to the previous value of I. Limit the number of times the mortgage calculation can be performed in any one run to 5. Add the following lines:

	Loop Characteristic:
`12 I=0`	**initialization**
`15 I=I+1`	**incrementation**

Change Line 50 to:

`50 IF I<5 GOTO 15`	**stop mechanism**

BEEP produces an audible signal the number of times indicated after the command. Insert **BEEP** 2: in the last line of the mortgage program to indicate completion.

```
70 BEEP 2: END
```

FOR . . . TO, STEP, NEXT

FOR . . . TO in combination with **NEXT** repeats an operation the specified number of times. **STEP** is added if the incremental value is other than one. A **FOR-NEXT** loop must begin and end with the same variable.

Anatomy of a FOR statement

Command	Followed by	Explanation
FOR	variable	index
=	expression	initial value of index
TO	expression	final value of index
STEP	expression	increment

For 30-year mortgages ranging from $10,000 to $100,000 at an interest rate of 15%, calculate the monthly payment. Increment by $10,000.

```
10 PAUSE "MORTGAGE PAYMENT"
20 I=15: N=30: i=I/12: n=N * 12
30 FOR V=10000 TO 100000 STEP 10000
40 PV=-V
50 WAIT 320: PRINT "PAYMENT $"; COMP PMT
60 NEXT V
70 BEEP 2: END
```

Run the program. The answers are:
$126.444, 252.888, 379.333, 505.777, 632.222, 758.666, 885.110, 1011.55, 1137.99, 1264.44

Nested looping is used if two or more variables are to be incremented. A **FOR-NEXT** loop must be completely inserted within another loop. Calculate the monthly payment on a 30-year mortage in the amounts of $30,000, $60,000, and $90,000 at interest rates of 14%, 15%, and 16%.

The program reads:

```
10 PAUSE "MORTGAGE PAYMENT"
20 n=30 * 12
30 FOR V=30000 TO 90000 STEP 30000
40 FOR I=14 TO 16
50 PV=-V: i=I/12
60 WAIT 320: PRINT "PAYMENT $"; COMP PMT
70 NEXT I
80 NEXT V
90 BEEP 2: END
```

Answer:

Mortgage Amount	Interst Rate	Monthly Payment
$30,000	14	$355.461
30,000	15	379.333
30,000	16	403.427
60,000	14	710.923
60,000	15	758.666
60,000	16	806.854
90,000	14	1066.38
90,000	15	1137.99
90,000	16	1210.28

USING, MDF

The answers to the mortgage problem have included more than two decimal places. **USING** formats an answer. It may be used by itself or with **PRINT**. A space must be left for the sign even if it is always positive. Since the answers for the mortgage problem were sometimes over $999, five spaces are required prior to the decimal point in the **USING** statement. After the decimal point, if only one digit would normally appear, a zero will be added. The third digit after the decimal point is truncated. Add to the mortgage program:

```
35 USING "#####.##"
```

Change Line 60 to provide room for the answer:

```
60 WAIT 320: PRINT "PAYM. $"; COMP PMT
```

If a particular USING format will be incorporated many times into a program, it is more efficient to use it as a string. At the beginning of the program include U$="####.##". In the program the statement becomes **USING** U$.

To avoid errors in a program eliminate a using format as soon as it is no longer needed by including the statement **USING** by itself.

The **USING** command can be placed within a **PRINT** statement. The complete format is:

PRINT (STRING, if any), **USING** (Format Expression); Variable
The comma is only required with a string.

As mentioned above, the **USING** statement will truncate after the state number of decimal points. In many business problems, rounding up and down is required. Inserting the **MDF** command before the variable will provide for rounding.

GOSUB

Subroutines are a powerful programming tool. A portion of a program which may be repeated many times need be entered only once as a subroutine. Long programs are easier to read and compicated portions of a program can be moved around without having to be entered each time.

To return from a subroutine the command is **RETURN**. In the mortgage program set up the financial key calculations as a subroutine:

```
10 PAUSE "MORTGAGE PAYMENT"
20 INPUT V,I,N
30 GOSUB 500
40 WAIT 320: PRINT "PAYMENT $"; COMP PMT
50 BEEP 2: END
500 PV=-V: i=I/12: n=N*12
600 RETURN
```

CLEAR, ERASE FIN, REM

CLEAR is used at the beginning of a program to insure that all variables are reset to zero and adequate space is cleared in memory for arrays. **CLEAR** also helps provide storage areas during a program when space is limited.

ERASE FIN will clear all of the financial memories which could have values retained from operation in the CAL mode.

REM stands for remarks. Use it to make notes in a program. Items under **REM** will not appear when the program is run, but will appear in the program list.

Add these three commands to the mortgage program:

```
10 CLEAR : ERASE FIN: PAUSE "MORTGAGE PAYMENT"
15 REM PUT IN VARIABLES
20 INPUT V,I,N
25 REM CALCULATE PAYMENT
        rest of program remains the same
```

DATA, READ, RESTORE

Two commands are necessary for placing data directly into a program: **DATA** and **READ**. The **DATA** statement can be placed at any point in a program before **END**. Each item in the statement is separated by a comma. The grouping of the data is controlled by the **READ** statement. **DATA** statements can be broken into several lines.

The **READ** statement reads the data in the order of the variables as listed within the statement.

Calculate the monthly payment for 3 mortgages:

The program is:

```
10 READ V,I,N
20 GOSUB 500
30 USING "#####.##"
40 PRINT "PAYMENT $"; MDF COMP PMT
50 GOTO 10
60 DATA 30000, 16, 30, 40000, 15.5, 25, 50000, 14.9, 22
70 BEEP 2: END
500 PV=-V : i=I/12: n=N*12
600 RETURN
```

Run the program. The answers are: $403.43, $527.90, $645.67

The EL-5510 will indicate an "out-of-data" error when all the data has been used. This error can be avoided by placing a "flag" in the program tied to an **IF . . . THEN** statement. The flag acts as a test of the data and should be a value significantly different in form the other data being read. For example, if all values are positive, –1 is a good flag. In the example above, add to the data line: –1, 0, 0 (null data must be included for all the variables):

```
25  IF V=-1 THEN BEEP 2: END
```

Eliminate Line 70.

A choice of different sets of data is the function of **RESTORE**. Three sets of mortgage data are used by a broker. The adjusted program allows the user to make a choice by indicating the number.

PRINCIPAL:	INTEREST RATE:	TERM:
30,000	14, 15, 16	30, 25, 20
40,000	15, 16, 16.5	30, 28, 25
50,000	15.5, 16, 17	35, 30, 28

Program:

```
 10  CLEAR: ERASE FIN
 20  INPUT "SET 1, 2, 3?"; A
 30  IF A=1 RESTORE 110: GOTO 60
 40  IF A=2 RESTORE 120: GOTO 60
 50  IF A=3 RESTORE 130: GOTO 60
 60  BEEP A: READ V,I,N
 65  IF V=-1 GOTO 140
 70  GOSUB 500
 80  USING "#####.##"
 90  WAIT 320: PRINT "PAYMENT $"; MDF COMP PMT
100  GOTO 60
110  DATA 30000, 14,30, 30000, 15,25, 30000, 16,20,-1,0,0
120  DATA 40000, 15,30, 40000, 16,28, 40000, 16.5,25,-1,0,0
```

```
130 DATA 50000, 15.5,35,50000, 16,30, 50000, 17,28,-1,0,0
140 END
```

The subroutine program remains the same.

ON

The **ON** statement provides a mean of combining several **IF . . . THEN** statements into one. The expression after **ON** has to equal a whole number. The expression is usually followed by **GOTO** or **GOSUB** and a series of line numbers. In an **ON** statement the computer will assume that the first of the line numbers is one, the second two, and so forth.

An example of **ON . . . GOSUB** appears in the Depreciation example (see Chapter on Applications).

DEF, AREAD

The letters on the lower two lines of the EL-5510 keyboard will bring forth specific programs when used to indicate a particular program. The use of **RUN** is eliminated. Define the mortgage program as **M** by changing the first program line to:

```
10 "M": CLEAR: ERASE FIN: PAUSE "MORTGAGE PAYMENT"
```

To use the program in the RUN mode press **DEF** and **M**.

A defined program can have the initial data entry made with the **AREAD** command. This command must appear in the first line of the program.

Calculate the monthly payment for 30-year mortgages at 15% for principal values of 35,000, 40,000, 70,000, and 80,000. Change the program to:

```
10 "M": AREAD V
20 I=15: N=30
30 GOSUB 500
40 USING "#####.##"
50 WAIT 320: PRINT "PAYMENT $"; MDF COMP PMT
```

```
60 BEEP 2: END
```

The subroutine remains the same.

Change to RUN mode.

Key in:		*Answer:*
35000	DEF M	$442.55
40000	DEF M	505.78
70000	DEF M	885.11
80000	DEF M	1011.56

DIM, CONCATENATION

An array is a set of values named by one letter with a changing subscript. The number of values in the array is called its dimension. The subscript is called the index and a two-dimensional array will have two indices.

The use of arrays on the pocket computer should be limited due to the relatively small memories of these devices. The command **DIM** is used to create an array. The maximum size of the index in an array is 255. Zero is counted in an index, e.g., **DIM** C$(3) sets up four reserved spaces. If no size is specified for an index, 16 characters are automatically reserved for each item. The maximum number of characters is 80. Specifying the size, the command is:

```
DIM C$ *20.
```

The user of the mortgage program wants to create a data base indicating the distance from the center of the city of five communities. Each item of data in the array will contain seven characters:

```
10 "A": DIM B$(4)*7
20 FOR I=0 TO 4
30 INPUT B$(I)
40 NEXT I
50 FOR I=0 TO 4
```

```
60 PRINT B$(I)
70 NEXT I
80 GOTO 50
```

To place the data into the computer go to the RUN mode and press DEF A. Each time ? appears type the entry and press **ENTER**. In this example:

```
? TROY-14    ENTER          ? HEMP-18   ENTER
? NCITY28    ENTER          ? NROC-15   ENTER
? ORNG-32    ENTER
```

The first data item appears automatically after the last entry. Each time **ENTER** is pressed, each item will be displayed on the screen.

Multidimensional **DIM** statements are created by adding commas and expressions within the parenthesis. An excellent example of this technique is contained in the "spreadsheet" program in the financial part of the Applications chapter.

Concatenation is the combining of two or more strings into a phrase. The symbol + is always required and " " provides spacing.

A real estate broker places standard descriptions into a **DIM** statement to be combined to describe a property, including:

Age: Older New
Bedrooms: 3-Bed 4-Bed 5-Bed
Type: Ranch Colonial

Program:

```
10 DIM B$ (6)
20 FOR I=0 TO 6
30 INPUT B$(I)
40 NEXT I
45 REM Dimensions A$(0) large enough to cover all combos.
50 DIM A$ (0)*25
60 A$(0)=B$(1)+” “+B$(2)+” “+B$(5)
```

```
70 PRINT A$(0)
80 END
```

Run the program keying in the 7 descriptive items. The values in Line 50 can be varied to meet each requirement. This A$ will read: NEW 3-BED RANCH.

AND, OR, NOT

AND, OR, and **NOT** are relational operators used with **IF . . . THEN** statements. Each is a logic test.

In the discussion of **IF . . . THEN** the following line was used:

```
50 IF PMT<500 THEN 20
```

a. Add the condition that the interest rate has to be less than 16%:

```
50 IF PMT<500 AND I<.16 THEN 20
```

b. Either the payment must be lower than $500 or the interest rate below 15%:

```
50 IF PMT<500 OR I<.15 THEN 20
```

c. **NOT** reverses a condition

```
50 IF NOT PMT<500 THEN 20  (same as P>500)
```

LEN, LEFT$, MID$, RIGHT$

LEN indicates the number of characters contained in a string.

```
10 Z$=“BOST, MA”
20 PRINT LEN (Z$)
```

Run the program. Answer is 7.

If a series of strings were to be analyzed by the first letter, add to the above program:

```
30 A$=LEFT$ (Z$, 1)
40 PRINT A$
```

After the answer of 7, press **ENTER**. The answer will be B.

To add a listing of the states of each city in a hypothetical directory, use **RIGHT$**. The number in the parenthesis indicates the number of digits from the right the letters should be read.

```
50 S$=RIGHT$ (Z$, 2)
60 PRINT S$
```

After the previous answer, press **ENTER** again. The answer is MA.

To alphabetize a series of strings a command that identifies internal letters in a string is valuable **(MID$)**. The **MID$** expression has three parts: string identifier; starting point; number of characters.

```
70 X$=MID$ (Z$, 2, 2)
80 PRINT X$
```

Press **ENTER** again. The answer is OS.

If a program user should answer "yes" or "no" to a question requiring a "y" or "n" reply, an error will occur. To reduce the chance of this type of error, the following program segment can be used:

```
50 INPUT "RERUN (Y/N)?"; Q$
60 IF LEFT$ (Q$, 1) = "Y" THEN (line #)
70 IF LEFT$ (Q$, 1) = "N" THEN (line #)
80 PAUSE "PLEASE ANSWER" : GOTO 50
```

Line 80 brings the user back to Line 50 if the answer to Line 50 does not begin with Y or N.

ASC, CHR$

Every character under the ASCII code has a decimal value. To count the number of times a letter occurs it is simpler to use the numeric value. To determine this value of the first letter of the string in the previous program:

```
90 PRINT ASC (Z$)
```

Pressing **ENTER** gives the value of 66.

In the RUN mode, key in:

```
CHR$ (66) ENTER
```

Answer: B

The **CHR$** statement converts an ASCII code number into its character equivalent.

STR$, VAL

STR$ turns a value into a string and **VAL** takes a value from a string. A number entered as a string will appear on the left side of the screen.

For a mortgage analysis it is necessary to produce the date of the last payment of the mortgage. The date is a string consisting of a three letter abbreviation for the month and a four-digit year. The following program adds the number of years of the term to the date.

```
10 INPUT B$
20 K = VAL RIGHT$ (B$,4)
30 INPUT "TERM="; T
40 L=K+T
50 Q$=LEFT$ (B$,3)
60 C$=Q$ + STR$ L
70 PRINT C$: END
```

If the mortgage begins in June of 1983 and runs for 30 years, the first input is JUN1983; TERM= is 30; and the final answer which will appear on the display is: JUN2013.

INKEY$

The **INKEY$** command is used to either speed up input or to limit the users choices. With **INKEY$** the input is limited to the key pressed and does not require the pressing of **ENTER**. The command takes the form:

```
10  A$ = INKEY$
```

RANDOM, RND

Random number generators primarily are used in statistical research and games. The **RANDOM** statement changes the beginning point for the generator each time to avoid duplication. The numbers are generated by the statement **RND** (*n*) with *n* being any number between 0 and 9.999999999 E 99.

A real estate office sold 200 houses during the past year. Select 10 sales at random to check the average sales level.

Program:

```
10  RANDOM
20  USING "####"
30  WAIT 150: PRINT RND (200)
40  GOTO 30
```

RUN. The first 10 numbers are:

```
84, 117, 89, 39, 78, 190, 151, 51, 153, 113
```

Try it. Do you get the same numbers?

PASS

The password system protects programs in the computer from inadvertent erasure or modification and from unapproved copying. To protect the program or programs in memory the **PASS** command is followed by a character string which cannot exceed seven characters. To protect the previous program call it **MORTPAY** and key in:

```
PASS "MORTPAY" ENTER
```

The **LIST** command will not produce the statements in the program. To remove the password, key in the identical statement.

Debugging, TRON, TROFF

Errors in a program are called "bugs". One way of checking a program for problems is to use the trace mode. The mode is activated by keying in **TRON** and **ENTER**. The line number of each statement is displayed after execution of the statement. The next statement will not be performed until the ↓ key is pressed. To check the statement just executed press the ↑ key. The trace mode remains in effect until **TROFF** is keyed in.

Tape Recorder Commands

With the tape cassette interface, programs and data can be saved on standard ormicrocassette size tape recorders.

The command **CSAVE** places a program on cassette tape. By itself, **CSAVE** will transfer all programs in the EL-5510 memory to tape. Placing a comma after **CSAVE** followed by a name in quotations will make that name a password for the file being transferred to tape.

To identify one program from another on tape each can be assigned a file name. To transfer with a file name:

```
CSAVE "NAME"
```

Save the earlier mortgage payment program on tape under the name MORT (**CSAVE** "MORT"). When a file name is used a password can also be added as previously described.

CLOAD? will check the program in memory against the program as stored on tape. An error signal signifies that the program should be loaded on tape again.

The **CLOAD** command clears the computer's memory and loads the first program from the tape. **CLOAD** followed by a file name in quotes will result in the tape being searched for that file name and loading only that specific program. **CLOAD** "MORT" will search for the mortgage program and load only it.

Data transfer to and from tape is accomplished with the statements **PRINT#** and **INPUT#**. To save the array in the **DIM** example on tape add the line:

```
PRINT# "DIST"; B$ (*)
```

Note: DIST is the file name. * indicates that all values of the variable are to be saved.

Printer Commands

The printer adds a new dimension to pocket computing. Printing the mortgage payment program output allows a banker or real estate agent to hand an actual customized document to a potential customer. The basic command is **LPRINT**. All print commands following the line **PRINT=LPRINT** will activate the printer. To switch back insert the line **PRINT=PRINT**.

The mortgage payment program is expanded below to include a print-out of the initial information and an optional amortization schedule. Each month:

B = Interest portion of payment = Remaining principal * Monthly interest rate = R * I

G = Principal portion of payment = Monthly payment – Interest portion of payment = P – B

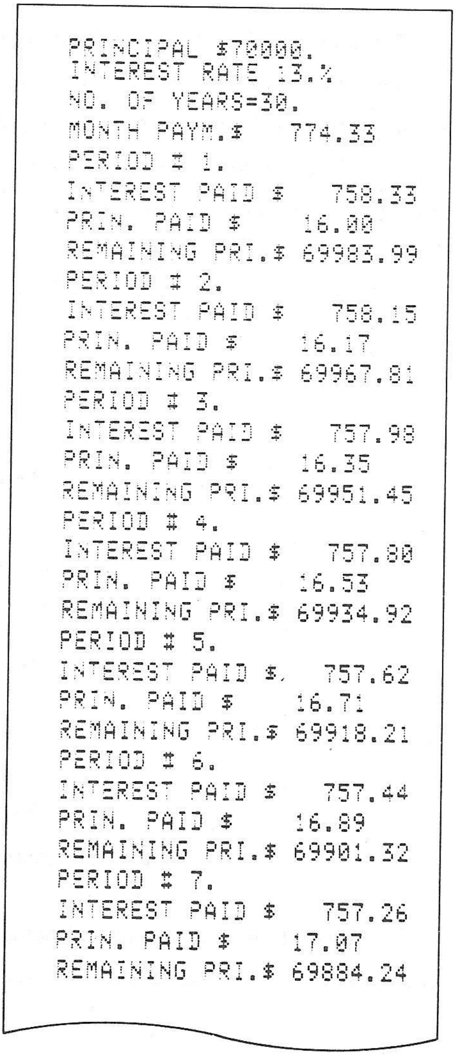
R = Remaining principal after payment = Previous principal – Principal portion of payment = R – G

Program:

```
10 “A”: CLEAR : ERASE FIN : PAUSE “MORTGAGE PAYMENT”:
   USING
15 REM INITIAL VALUES
20 INPUT “V=”; V, “I=”; I, “N=”; N
25 REM CHANGE I & N TO MONTHLY BASIS, CALCULATE
   PAYMENT
30 i=I/12: n=N*12: PV=-V: P=COMP PMT
40 PRINT = LPRINT
50 PRINT “PRINCIPAL $”; V
60 PRINT “INTEREST RATE ”; I; “%”
70 PRINT “NO. OF YEARS=”; N
80 USING “######.##”
90 PRINT “MONTH PAYM. $”; MDF P
100 USING
110 INPUT “FULL SCH. (Y/N)?”; C$
120 IF C$=“Y” THEN 200
130 END
200 REM INITIAL PRINCIPAL VALUE IS V
210 R=V
220 J=J+1
230 PRINT “PERIOD # ”; J
240 AMRT J
250 PRINT “INTEREST PAID $” ; USING “#####.##” ; MDF INTE
260 PRINT “PRIN. PAID $”; MDF PRN
270 PRINT “REMAINING PRIN.$”; MDF BAL
280 REM TEST NUMBER OF PERIODS
290 IF J<n THEN 220
300 END
```

Run the program by pressing **DEF** A in the RUN mode.

	Key in:
V =	70000
I =	13
N =	30
FULL SCH. (Y/N)	Y



```
PRINCIPAL $70000.
INTEREST RATE 13.%
NO. OF YEARS=30.
MONTH PAYM.$   774.33
PERIOD # 1.
INTEREST PAID $    758.33
PRIN. PAID $     16.00
REMAINING PRI.$ 69983.99
PERIOD # 2.
INTEREST PAID $    758.15
PRIN. PAID $     16.17
REMAINING PRI.$ 69967.81
PERIOD # 3.
INTEREST PAID $    757.98
PRIN. PAID $     16.35
REMAINING PRI.$ 69951.45
PERIOD # 4.
INTEREST PAID $    757.80
PRIN. PAID $     16.53
REMAINING PRI.$ 69934.92
PERIOD # 5.
INTEREST PAID $    757.62
PRIN. PAID $     16.71
REMAINING PRI.$ 69918.21
PERIOD # 6.
INTEREST PAID $    757.44
PRIN. PAID $     16.89
REMAINING PRI.$ 69901.32
PERIOD # 7.
INTEREST PAID $    757.26
PRIN. PAID $     17.07
REMAINING PRI.$ 69884.24
```

CHAPTER 4

APPLICATIONS

Part 1 — Business

A. Break-Even Analysis

"Break-even" is a business technique used to determine the quantity of a product where total costs will equal variable costs plus fixed costs. Fixed costs are depreciation, light, power, utilities, rent, insurance, administrative salaries and others. Variable costs are materials and labor. The numberof units required to reach the break-even point is:

$$\text{Quantity} = \frac{\text{Fixed Cost}}{\text{Price} - \text{Variable cost/unit}}$$

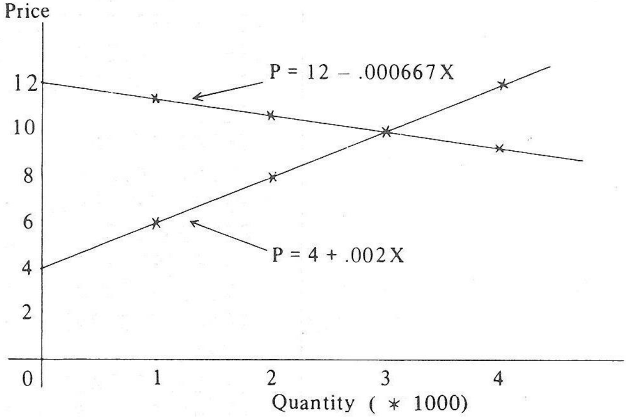


Figure 4.2 Supply-Demand Curves

It is possible to use supply and demand curves to determine the price. Supply curves measure the willingness of manufactureers to provide product at different prices. Demand curves reflect the willingness of consumers to buy at different prices.

The supply and demand curges can alternatively be used to determine the price in the break-even equation. The basic equation for these curves is P = A + BX.

F = Fixed Cost
P = Price
V = Variable cost/unit

Program:

```
10 "A": CLEAR ; PAUSE "BREAK-EVEN ANAL"
20 PAUSE "PRICE CALC"
30 INPUT "A ="; A, "B="; B
40 INPUT "F.C.="; F, "V.C.="; V
50 INPUT "X="; X
60 P=A+B*X
70 PRINT "P= ";P
80 REM BREAK-EVEN ANALYSIS
90 W=F/(P-V)
100 Z=W-X
105 USING "#####."
110 PRINT "AMT.="; W
115 PRINT "DIFF.=";Z
120 IF Z<-1 OR Z>1 THEN GOTO 50
130 END
```

The supply curve for a particular product is: P = 4 + .002X. The demand curve is: P = 12 – .000667X. The fixed cost is $10,000 and the variable cost is $6.00. Determine break-even quantities based on the supply and demand curves. Use supply and demand quantities of 2,000 in each case.

Change to RUN Mode

Key in: **DEF A**	**NOTES**
A = 4 **ENTER**	Using Supply Curve
B = .002 **ENTER**	
F.C. = 10000 **ENTER**	
V.C. = 6 **ENTER**	
X = 2000	Quantity
Answer:	
P = $ 8 **ENTER**	Price
AMT. = 5000	Break-even Quantity
Diff. = 3000	Difference between break-even quantity and supply curve quantity.
Key in: **DEF A**	
A = 12 **ENTER**	Using demand curve
B = –.000667 **ENTER**	
F.C. = 10000 V.C. = 6 X = 2000	
Answer:	
P = 10	Price
AMT. = 2143 DIFF = 143	Difference between break-even quantity and demand curve quantity.
ENTER	
X = 3000 **ENTER**	
P = 9	
AMT. = 2500 DIFF = –499	Demand quantity at $10 is 500 more than required to break-even.
Retry with Supply Curve	
Key in: **DEF A**	
A = 4 B = .002 F.C. = 10000 V.C. = 6	
X = 3000 **ENTER** (New Value)	

Answer:

P = 10 AMT = 2500 DIFF = –500

(same answer as for Demand Curve)

This is the equilibrium point of the two curves. When the equilibrium price is higher than required to break even, the investment situation is favorable.

Purchase vs. Lease

"To buy or to lease", that is the question. Is it more economical to own equipment or to lease it? This program will help to capture and to analyze most of the variables that influence the purchase versus lease decision.

With leasing, the arrangements are very different from purchasing:

1. No down payment is required
2. A business can "write off" (reduce tax liabilities) on the full lease payments.
3. There is no equity value, except for buy-backs which are not considered here.

When equipment is purchased:

1. A down payment is required — funds are committed that otherwise might be invested and earning a return.
2. The balance of the purchase may be financed — principal payments are not tax deductable.
3. The business owner may claim an Investment Tax Credit and depreciation to reduce tax liabilities.
4. The owner of the equipment has equity — potential income from resale.

That program below assumes that:

1. Payments are made annually
2. Number of Lease Payments = Estimated Economic Life
3. There is no investment tax credit

Program Notes:

A. Payments to the lessor or the bank are taken as positive values; therefore, the

Net Present Values will be negative. Line 380 of the program converts these values to positive.

B. If payments are to be made at the beginning of the period **BGN** is turned on by a Y reply on Line 45. Payments to the bank are assumed to be made at the end on the period and **BGN** is automatically turned off as the program passes line 200.

C. The cash outlay is divided by 2 on Line 70 to reflect a 50% tax bite.

D. Line 90 shows the procedure for inputting the same cash flow amount over a specified number of periods. For uneven cash flow amounts the **INPUT** command is added.

E. **ERASE FIN** is used on Line 200 to clear all of the financial registers as they are used again in the purchase calculation.

F. The initial cash flow in a purchase situation is the down payment which is on Line 250.

G. The **FOR . . . NEXT** sequence that follows places the cash flow amounts in for the periods in which the loan is repaid. The cash flow for the remaining periods are covered in the next state of the program. These last amounts are negative because they are not payments to the bank.

H. The last cash flow amount has to include the salvage value and this amount is calculated on Line 350.

Program:

```
10  CLEAR :ERASE FIN: PAUSE "PURC. VS. LEASE"
20  INPUT "CAP. INV."; C
30  REM LEASE CALCULATION
40  INPUT "INT. RATE"; I, "TERM"; Y
45  INPUT "BGN (Y/N)"; Q$
50  REM ANNUAL NET CASH OUTLAY
60  i=I : n=Y : PV=C : B=MDF COMP PMT
70  T=B/2
80  FOR N=1 TO Y
```

```
90  CFI (N)=T:NI (N)= n
100 NEXT N
110 X=MDF COMP NPV
120 WAIT 420: PRINT “NPV-LSE”, X
200 ERASE FIN : BGNOFF
210 REM PURCHASE CALCULATION
220 INPUT “PAYMENT”;P, “INT.RATE”;I, “FIN.PER.”;N,
    “SALV.VAL”;V, “DISC.RATE”; F
230 PV=C-P: i=I:n=N:D=C/Y
240 W= MDF COMP PMT
250 CFI (0)=P
260 FOR Z=1 TO N
270 AMRT Z:L=MDF PRN : J=W-L:S=.5 * (J+D)
280 REM J=INTEREST PAID ** S=TAX SAVINGS
290 CFI (Z)=W-S
300 NEXT Z
310 REM REMAINING CASH FLOW PERIODS
320 FOR R=Y-N TO Y-1
330 CFI (R)=-.5 * D
340 NEXT R
350 CFI (Y)=-.5 * D-V: i=F
360 H=MDF COMP NPV
370 WAIT 420: PRINT “NPV-PUR”, H
380 IF -H>-X GOTO 400
390 PRINT “LEASE BEST!”: END
400 PRINT “**PURCHASE**”: END
500 BGNON: RETURN
```

EXAMPLE:

A new computer for the BNW Company will cost $60,000. The equipment has a life of 8 years and if leased will be paid for in 8 annual installments. The percentage for a lease is 10.5% and payment is due upon delivery and at the beginning of each subsequent year.

A loan from the bank will be at 10% for five years with 75% financing. Payments are made at the end of each year. The after tax cost of capital for the company is 9% (discount factor).

Should the company purchase or leas this equipment?

Depreciation

The assets of a business (vehicles, equipment, buildings, etc) are "used up" or worn out over a period of years causing a decrease in value. This decrease in the value of an asset is called depreciation. Depreciation is usually based on historic costs (the actual cost of an item). Current or replacement costs in most cases will be higher and consequently the depreciation taken is too low. Depreciation lower than replacement costs causes the overstatement of the current value of capital assets, earnings, and taxes.

Accelerated depreciation is used to compensate for the replacement cost problem. Changing the rate of depreciation does not change cash flow, except if taxes are considered.

The continuation of increasing costs due to inflation in the indefinite future strengthens the case for accelerated depreciation. The incentive to increase cash flow in the short term is greatest when the revenues available this year will buy more than the same amount of revenues generated one, two or three years later.

The most commonly used techniques of depreciation are straight line, declining balance, and sum-of-years' digits.

To depreciate an asset using straight-line depreciation (a constant decrease in value, or constant-balance method) requires establishing the current value (C), the salvage or residual value (S), and the life of the asset in years (Y).

Formula for Straight-Line Depreciation:

$D = (C - S) / Y$

C = Current Value
S = Salvage Value
Y = No of Years

Straight-line depreciation has been altered by the IRS tax codes and is called the Accelerated Cost Recovery System (ACRS):

3-year property	
1st year	25%
2nd year	38%
3rd year	37%

5-year property	
1st year	15%
2nd year	22%
3rd – 5th	21%

10-year property	
1st year	8%
2nd year	14%
3rd year	12%
4th – 6th	10%
7th – 10th	9%

ACCELERATED DEPRECIATION

The accelerated depreciation (declining-balance) method requires specifying the initial book value or current book value (C), the life of the asset in years (Y), and the acceleration factor (A). Double declining or 200% is the most common acceleration factor.

The formula for accelerated depreciation of a business asset is shown below. The book value (C) is multiplied by the acceleration factor (A) and then divided by the life of the asset in years (Y).

Year 1 $D1 = C * A / 100 / Y$
Year 2 $D2 = (C - D1) * A / 100 / Y$
Year 3 $D3 = (C - (D1 + D2)) * A/100/Y$
Etc. . . . .

D = Depreciation
C = Book Value
A = Acceleration factor (%)
Y = Years

SUM OF THE YEAR'S DIGITS DEPRECIATION

Another method for computing accelerated depreciation is called sum of the year's digits. With this method an even faster "write off" of an asset's value may be obtained initially. This method requires a salvage value, thus improving the "Balance Sheet" with respect to assets and net worth during the later years.

To computer depreciation using this method, the asset's value or remaining value is multiplied by a ratio of the number of years remaining divided by the sum of the years of the asset's life. For example, if an asset has a six year life, the sum is represented by:

$$6 + 5 + 4 + 3 + 2 + 1 = 21$$

The formula is:

Year 1 $D1 = C * (Y/S)$
Year 2 $D2 = (C - D1) * (Y / S)$
Year 3 $D3 = (C - (D1 + D2)) * (Y / S)$
Etc.

D = Depreciation
C = Current Book Value
Y = Remaining Years of Life
S = Sum of the Year's Digits

Program Notes

1. All results are rounded to the nearest dollar by using +.5 with INT.
2. LINE 25 Keeps the value of C at its original amount to avoid errors if each subroutine is used.

Program:

```
10  CLEAR : PAUSE "DEPRECIATION"
20  INPUT VALUE";C , "SALVAGE"; S
25  B = C
30  PAUSE "CHOOSE DEP. TYPE"
```

```
40   WAIT 50: PRINT “1 = STRAIGHT-LINE”
50   PRINT “2=ACRS”
60   PRINT “3=DOUBLE DECLINE”
70   PRINT “4=SUM–YEARS”
80   WAIT: INPUT “ENTER 1–4”; Z
90   IF Z> 4 OR Z<1 THEN 30
100  BEEP Z: ON Z GOSUB 200, 300, 500, 700
120  INPUT “RECALCUL. (Y/N)”; X$
130  IF X$=“Y” GOTO 30
140  END
200  REM STRAIGHT-LINE
210  INPUT “#YEARS”; Y
220  PRINT “DEP/YR=”, INT (((C–S)/Y) + .5)
230  RETURN
300  REM ACRS DEPRECIATION
310  INPUT “#YEARS (3,5,10)”; Y
320  IF Y=5 GOTO 370
330  IF Y=10 GOTO 400
340  PRINT “YEAR1=”, INT (.25 * C+.5)
350  PRINT “YEAR2=”, INT (.38 * C+.5)
360  PRINT “YEAR3=”, INT (.37 * C+.5) : RETURN
370  PRINT “YEAR1=”, INT (.15 * C+.5)
380  PRINT “YEAR2=”, INT (.22 * C+.5)
390  PRINT “YR 3, 4, 5=”, INT (.21 * C+.5): RETURN
400  PRINT “YEAR1=”, INT (.08 * C+.5)
410  PRINT “YEAR2=”, INT (.14 * C+.5)
420  PRINT “YEAR3=”, INT (.12 * C+.5)
430  PRINT “YR 4, 5, 6=”, INT (.1 * C+.5)
440  PRINT “YR 7–10=”, INT (.09 * C+.5)
450  RETURN
500  REM DOUBLE DECLINING
```

```
510  INPUT "#YEARS"; Y
520  F = 2 / Y
530  FOR I = 1 TO Y
540  D = C * F
550  PRINT "YEAR "; I; "="; INT (D+.5)
560  C = C - D
570  NEXT I
580  RETURN
700  REM SUM-OF-YEARS
710  INPUT "#YEARS"; Y, "SUM-OF-YEARS"; W
720  FOR I = Y TO 1 STEP -1
730  PRINT "YEAR "; X; "="; INT ((B-S) * (I/W) + .5)
740  X = X + 1
750  NEXT I
760  RETURN
```

EXAMPLE:

The depreciable fixed assets required for a proposed box manufacturing production line consists primarily of forming machinery. The machinery has a cost of $100,000, a useable life of five years and a salvage life of $20,000. All the techniques of depreciation will be compared.

Run the program. You should get the following results:

Straight-Line: $16,000/Year

Year #	**ACRS**	**Double-declining**	**Sum-of-the-years**
1	$15,000	$40,000	$26,667
2	22,000	24,000	21,333
3	21,000	14,400	16,000
4	21,000	8,640	10,667
5	21,000	5,184	5,333

Moving Averages

A moving average takes the average of the prior data points and uses the average as the smoothed result. Smoothing is necessary in many types of business and economic studies to eliminate unusual single data points which would tend to obscure the overall trend.

Programs are presented for a 3-point moving average, a weighted 3-point, and for a moving average with the number of data points specified by the user.

3-point:

```
10  CLEAR
20  DIM D(2)
25  REM Initial Data Input
30  INPUT D(2), D(1), D(0)
40  GOTO 70
50  D(2)=D(1): D(1)=D(0)
60  INPUT "NEW D", D(0)
70  W=(D(0)+D(1)+D(2))/3
80  PRINT "AVE.=";W: GOTO 50
90  END
```

Example:

Data: 1, 2, 3, 6, 12, 21

```
Key in:                       Display:
  RUN    ENTER                ?
   1       "                  ?
   2       "                  ?
   3       "                  AVE.=2
           "                  NEW D
   6       "                  AVE.=3.666666667
           "                  NEW D
  12       "                  AVE.=7
```

```
               "                NEW D
   21          "                AVE.=13
```

3-point Weighted:

Change Line 70 in the above program to:

```
70 W = (3 * D (0) + 2 * D(1) + D(2)) / 6
```

The answers for the above data will be:

2.333, 4.3333, 8.5, 15.5

N-point Moving Average:

Programming Notes:

1. The variable Q represents the actual number of points. N has to be one less because O is counted in a DIM statement. The value of Q is used again in calculating the average amount.
2. On Line 130 the value of B is restored to the value of N and R is reinitialized at O.
3. Where > and = are used together the > comes first.

```
 10 CLEAR: INPUT "#POINTS", Q: N=Q-1
 20 DIM D(N)
 25 REM Input Initial Data
 30 FOR I=0 TO N
 40 INPUT D(I)
 50 NEXT I
 60 GOTO 100
 70 D(B)=D(B-1)
 80 B=B-1: IF B>=1 GOTO 70
 90 INPUT "NEW D", D(0)
100 FOR I=0 TO N
110 W=D(I)/Q : R=R+W
120 NEXT I
```

```
130 PRINT "AVE.="; R: R=0: B=N: GOTO 70
140 END
```

A common use for this type of moving average is 12 points for data compiled on a monthly basis.

Part 2 — Financial

FUTURE VALUE OF A SERIES OF PAYMENTS

All of the applications of the Financial Keys discussed in Chapter 2 may be placed in programs. Programming these applications increases the ease with which the user can carry out the procedure without having to consider the process.

The future value of a series of payments is an example. The programs asks the user to place into the computer each item needed to carry out the calculation.

Program:

```
10   CLEAR: ERASE FIN
20   INPUT "DEPOSIT"; D, "PAYMENT";P, "INT. RATE";I
     "YEARS";Y, "PAY/YR";W
30   REM CALCULATE INPUTS
40   PV = –D: PMT = –P: i = I/W: n = Y * W
50   USING "#####.##"
60   PRINT "FUT.VAL="; MDF COMP FV
70   END
```

Example:

With an initial deposit of $200 and payments of $50 each quarter, how much will accumulate at the end of 3 years at 52%?

Answer: $879.14

CONSTANT PRINCIPAL LOAN

Amortization of a loan based on an equal payment each period is described in Chapter 3. Some loans require each payment to include an equal amount for principal. The payment will change each period.

This example must be used with the CE-126 interface. Users without the interface should replace any **LPRINT** with **PRINT**.

Program:

```
 10  CLEAR: ERASE FIN: LPRINT ”MONTHLY PAYMENT”
 20  LPRINT  “ EQUAL PRIN/MON”
 30  INPUT   “MONTH =”; E, “DAY =”; D, “YEAR =”; H
 40  INPUT   “PRINC. =”; P
 50  INPUT   “INT. =”; I
 60  INPUT   “YEARS =”; Y
 70  LPRINT  “DATE: ”; E; D; H
 80  LPRINT  “12 PAYMENTS/YEAR”
 90  LPRINT  “PRINC. $”; P
100  LPRINT  “INT. RATE”; I; “%”
110  LPRINT  “TERM =”; Y; “YEARS”
120  n=Y*12: B=P: X=P/n: i=I/12
130  K=E
140  C=B*i
150  M=X+C: B=B–X
160  L=L+1: K=K+1
170  IF K=1 THEN LET H=H+1
180  PRINT = LPRINT
190  PRINT “#”; L; “   ”; K; “   ”; K; D; H
200  USING “######.##”
210  PRINT “INT.=$”; C
220  PRINT   “PRIN.=$”; X
```

```
230  PRINT  “PAYM. = $”; M
240  PRINT  “BAL. = $”; B
250  IF K = 12 THEN LET K = 0
260  USING
270  IF L>n THEN GOTO 140
280  END
```

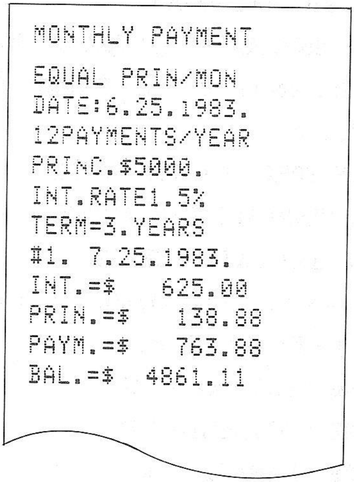
Calculate the monthly payments for a $50,000 loan for 3 years at 15%.

Change to **RUN** Mode

Key in:

MONTH = 6   DAY = 25   YEAR = 1983   PRINC = 5000   INT. = 1.5
YEARS = 3

Answer:



```
MONTHLY PAYMENT

EQUAL PRIN/MON
DATE:6.25.1983.
12PAYMENTS/YEAR
PRInC.$5000.
INT.RATE1.5%
TERM=3.YEARS
#1. 7.25.1983.
INT.=$    625.00
PRIN.=$    138.88
PAYM.=$    763.88
BAL.=$  4861.11
```

Rule of 78

Interest rebates on consumer loans paid in full before the end of the loan period are determined by a technique known as the Rule of 78. It is based on the remaining periods as a ratio of the sum of the digits of the total period. The basic equation is:

$$\text{REBATE} = \frac{(n - K + 1)\ (n - K)}{(n + 1)\ (n)}\ \text{FC}$$

Where:

FC = Total Finance Charge
n = number of periods
K = Installment that the loan is paid off on.

Program Notes: The program takes advantage of the special amortization keys on the EL-5510. Line 50 and Line 60 accumulate the total principal paid and the total finance charge.

Program:

```
10 CLEAR : ERASE FIN
20 REM ORIGINAL LOAN
30 INPUT “INT. RATE”; I, “YEARS”; N, “PRINC.”; P
40 i=I/12: n=N*12: PV=P: T=MDF COMP PMT
50 ACC 1,n
60 P=MDF SPRN: F=MDF SINTE
70 INPUT “INSTALL. REPLAY”; K
80 REM CALCULATE REBATE
90 R=((n-K+1)*(n-K)/((n+1)*n))*(-F)
100 D=-T*K: G=-F-P
110 PAUSE “TOT.PAY.=”, G
100 PAUSE “TOT.PAID=”, D
130 PAUSE “REBATE=”, R
140 PAUSE “PREPAY AMOUNT=”
150 PRINT MDF G-D-R
160 END
```

The BNW Company has a 5-year installment loan for \$100,000 at 12%. The loan is paid off at the end of the eighth month. What is the rebate and the net payment to prepay in full?

Answers: The rebate is \$25,200.50 and the payment is \$90,470.77.

IRA Analysis

An Individual Retirement Account (IRA) is a trust created by the taxpayer to provide retirement income. The maximum amount that can be contributed by an individual is $2000 each year. This contribution is free of federal income tax until it is actually used. The interest accumulations are treated in the same way.

If withdrawals are made prior to age 59 1/2, an excise tax of 10% is charged on the amount withdrawn.

The program for analyzing the IRA accumulation compares this amount to a Non-IRA accumulation at a specified point in time. It is assumed that the maximum contribution of $2000 is made each year and steady interest is received at a rate of 10% under each plan.

For comparison purposes, the IRA sum is reduced by the tax rate used at the end of the period. The interest accumulated on the Non-IRA plan is also reduced by the same amount. If the percentage shown at the end of the program is less than 100%, it will not be advisable to establish an IRA account. If the number of years of IRA accumulation added to the taxpayer's age is less than 59 1/2, then the accumulated amount is reduced by 10%.

The last part of the program takes the accumulation before the tax equalization and calculates the number of years the accumulated IRA amount can provide a specified income.

Program:

```
10   CLEAR: ERASE FIN
20   P=2000: I=10: BGNON
30   INPUT "TAX RATE(%)"; T, "YEARS"; N
40   REM IRA CALCULATION
50   n =N *12: i =I /12: PMT=-2000/12
60   S= COMP FV: V=S
```

```
70   INPUT "PENALTY (Y/N)"; Q$
80   IF Q$="Y" GOSUB 500
90   S=S*(1-T/100)
100  PAUSE "NET IRA ACCUM="
110  PRINT S
200  REM NON-IRA CALC.
210  X=P*(1-T/100): PMT =-X/12
220  Y= COMP FV: B=X*n/12
230  Z=B+(1-T/100)*(FV-B)
240  PAUSE "NON-IRA ACCUM="
250  PRINT Z
260  PAUSE "IRA:% OF NON-IRA"
270  PRINT INT (S/Z*100+.05); "%"
280  INPUT "CAL.INC.YRS (Y/N)"; R$
290  IF R$="Y" GOTO 320
300  END
320  INPUT "ANNUAL INC."; A
330  PV=V: PMT=A/12: COMP n
340  PRINT "YEARS=", -n/12: END
500  S=.9*S: RETURN
```

EXAMPLES:

1. At a tax rate of 30% and for a period of 20 years, compare the two methods of accumulation. If the taxpayer withdraws $10,000 each year after retirement, how many years will the trust last?

Key in: `RUN`

Display:	Enter:
TAX RATE	30
YEARS	20
PENALTY (Y/N)	N
NET IRA ACCUM =	
89331.31	
NON-IRA ACCUM ="	
70931.91	
126%	(26% greater accumulation with IRA account)
CAL. INC. YRS (Y/N)	Y
ANNUAL INC. =	10000
YEARS = 30.01	

If you wish the trust to last for only 20 years, what will the annual income be?

Key in: `n = –20 *12 ENTER COMP PMT ENTER *12 ENTER`

Answer: $12259

2. Will it pay to accumulate money in an IRA for 5 years, if the penalty will apply?

Key in: Tax Rate = 30 Years = 5 Penalty = Y

Answers: IRA Account = 8198.65 Non-IRA Acct. = 8476.72
Ratio = 97% It will not pay to create an IRA (ratio is less than 100%).

Part 3: Statistical Tests

STATISTICS

A. "T" Test

Statistical tests, as described in this section, are helpful in a variety of business settings. Certain data, when it is needed, is often too expensive to gather in large samples or is difficult to obtain from destruct-test results. Reliable results can be obtained with samples smaller than 30 by using the "t" test.

The t-distribution is close to the normal distribution and permits checking of a hypothesis even with a small sample.

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$

where $\bar{x}$ = sample mean
s = sample standard deviation
n = sample size
μ_0 = required population mean

The EL-5510 is particularly convenient for running "t" tests. Data input in the calculator mode is automatically stored in letter memories in the BASIC modes as follows:

LETTER	STATISTICAL DATA STORED
U	Σy^2
V	Σy
W	Σxy
X	Σx^2
Y	Σx
Z	n

The "t" test can be programmed using the store memories.

Let A = μ_0

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}} = \frac{\frac{\Sigma x}{n} - \mu_0}{\sqrt{\frac{\Sigma x^2 - n(\Sigma x/n)^2}{n(n-1)}}} = \frac{Y/Z - A}{\sqrt{(X - Z(Y/Z)^2)/Z(Z-1)}}$$

Program:

```
10    PAUSE "T-TEST"
20    INPUT "POP.MEAN="; A
30    T = (Y/Z-A) / √((X-Z*SQU(Y/Z))/Z*(Z-1))
40    USING "##.###^"
50    PRINT "T="; T
```

A sample of 5 transistors are used until they fail. The life (hours) of each is 7400, 8000, 8300, 8700, 9900. A buyer of transistors requires a mean-life of 8100 hours. Will the buyer be safe with a 5% level of confidence if these transistors are purchased?

Change to Calculator Mode:

Key in: **STAT**

7400 **DATA** 8000 **DATA** 8300 **DATA** 8700 **DATA** 9900 **DATA**

Change to RUN Mode

Key in: **RUN**

POP. MEAN = 8100 **ENTER**

Answer: T = 2.153 E – 01

From the table, the value for t at a 5% confidence level is 2.13. We use the sample size (n) –1 as the degrees of freedom.

Since the t calculated at .2153 is less than 2.13, the buyer can conclude that the transistors purchased could be from a population with a mean lifetime equal to or greater than 8100 hours 95% of the time.

B. Chi-square Test

The x^2 Test is used to determine if a set of data when compared to an ideal or predetermined distribution of that data has a variance from probability or some preset expectation. The formula for the test is:

$$x^2 = \Sigma \frac{(O - E)^2}{E}$$

where O = Observed Frequency
E = Expected Frequency

The hypothesis is rejected if the x^2 statistic falls in the rejection region of the x^2 distribution.

The proportion of adult females (over age 21) in society is 52% of total adults. The number of female workers earning above $20,000/year in three companies was established. Analyzing the data, is the difference between the actual proportions compared to the expected frequency due to chance.

EXPECTED FREQUENCY OF EMPLOYEES BY SEZ

RUN Mode

Key in: A = .52

COMPANY	TOTAL	FEMALE	MALE
A	10000 * A **ENTER**	5200	4800
B	5650 * A **ENTER**	2938	2712
C	3290 * A **ENTER**	1711	1579

The table below indicates the actual number with the expected below in parenthesis. Each cell will be calculated in the x^2 calculation.

	COMPANY		
	A	B	C
Female	3100	2900	1800
	(5200)	(2938)	(1711)
Male	6900	2750	1490
	(4800)	(2712)	(1579)
Total	10000	5650	3290

Program:

```
10   CLEAR: L=0: PAUSE "CHI–SQUARE"
20   INPUT "CELLS ="; N
30   INPUT "OBS ="; O, "EXP ="; E
40   G = SQU (O–E)/E
50   H=G+H: L=L+1
60   IF L<N THEN 30
70   USING "#####.##"
80   PRINT "CHI–SQU= "; H
90   END
```

Change to RUN Mode

Key in:

CELL = 6 **ENTER**

OBS = 3100 2900 **ENTER**

EXP = 5200 2938 **ENTER**, etc.

Answer: CHI–SQU = 1777.49

The $x^2_{0.05}$ column in the table represents the dividing line which cuts off 5% of the right tail of the distribution. The degrees of freedom is one less than the number of sample proportions being tested.

In the above example the *df* is 2 and $x^2_{0.05} = 5.991$. The value obtained is significantly higher indicating the difference between what was expected and what was actually happening was due factors other than chance.

Part 4 — Mathematics

The applications presented in this section are examples of the more advanced type of mathematical analysis that can be performed with the EL-5510.

A. Feet/Inches Conversions

Adding or subtracting dimensions given as feet and inches can be difficult and confusing. In many applications it is necessary to convert dimensions given in feet and inches to decimal.

The program converts all dimensions given in feet to their equivalent in inches. Entries in inches only require pressing **ENTER** when the prompt is for “Feet”. Inches including fractions can be entered by using the following format 5 + 5 / 8. When the prompt is for “Feet”, the total can be obtained by inputting =.

The answer appears on the display alternatively as decimal feet and as feet and decimal inches several times. The user can change the number of times the answers appear and the length of time on the screen to meet their particular preference.

Program:

```
10 “F”: CLEAR: U$= “###.####”
20 A=0: B=0: R$=“F”: S$=“IN”: T$=“=”: L=12
30 INPUT “ENTER FT.”;F$: IF F$=T$ THEN 70
40 A=VAL (F$): BEEP 1: INPUT “ENTER IN.”; B: BEEP 2
50 C=A*L+B+C: E=(A*L+B)/L
60 WAIT 100: PRINT A; R$; B; S$; T$
65 PRINT USING U$; E; R$: BEEP 3: USING : GOTO 30
70 IF C<0 LET C=–(ABS C): E=–C/L: A=–(INT(C/L)): B=–(C–
   (A*L): GOTO 90
```

```
80 E=C/L: A=INT (C/L): B=C−A*L
90 FOR K=0 TO 3: PRINT USING ;“TOTAL ”; A; R$; B; S$; T$
100 PRINT “TOTAL”; T$; USING U$; E; R$: NEXT K
120 INPUT “REV. TOTAL (Y/N)?”; Q$
140 IF LEFT$ (Q$, 1)=“Y” THEN 90
150 IF LEFT$ (Q$, 1)=“N” THEN 10
160 PAUSE “PLEASE ANSWER”: GOTO 120
```

Add 8′3 1/4″ to 11″ to 10′8 1/4″.

Change to RUN Mode.

Key in:	**DEF F**
8	**ENTER**
3+1/4	**ENTER**
	ENTER
11	**ENTER**
10	**ENTER**
8+1/4	**ENTER**
=	**ENTER**

Answer: 19.875 Feet or 19 Feet 10 1/2 inches

B. Quadratic Equations

The quadratic formula is used to solve quadratic equations when the roots cannot be factored. The equation to be solved is put in the form:

$$ax^2 + bx + c = 0$$

The solution of the roots takes the form:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

The portion of the root equation under the square root sign, $b^2 - 4ac$, is called the discriminant. When the discriminant is positive, there are two roots. A zero value for the discriminant results in a negative root and a negative discriminant produces roots which are complex numbers.

A complex number is shown as $a + bi$. To solve for complex roots, the negative sign is removed from the discriminant and i is added to the second part of the answer. The value of i is $\sqrt{-1}$.

Program:

```
10 “Q”: CLEAR: REM QUADRATIC EQUATION
20 INPUT “A=”; A, “B=”; B, “C=”; C
30 D=SQU B -4*A*C: IF D<0 THEN 120
50 PAUSE “ROOT 1=”
60 PRINT (√D -B)/2*A
70 PAUSE “ROOT 2=”
80 PRINT (-B-√D)/2*A
90 INPUT “AGAIN (Y/N)?”; Q$
100 IF Q$=“Y” THEN 10
110 END
120 REM ** FIRST PART/SECOND PART**COMPLEX NUMBER
130 N=-B/2*A: I=√ABS D/2*A
135 USING “###.###”
140 PAUSE “ROOTS=”
150 PRINT N; “;”; I; “I”: GOTO 90
```

Calculate the roots of the following equations:

a. $2x^2 + 8x - 10$

b. $x^2 - .2x + .004$

c. $20x^2 + 9x + 3897$

d. $x^2 + x + 1$

Key in:		DEF Q	Answer:
a.	2	ENTER	
	8	ENTER	
	–10	ENTER	ROOT 1 = 1.0
			ROOT 2 = –5.0
b.	1	ENTER	
	–.2	ENTER	
	.004	ENTER	ROOT 1 = 0.17746
		ENTER	ROOT 2 = 0.02254
c.	20	ENTER	
	9	ENTER	
	3897	ENTER	ROOTS = –0.225 ± 13.957i
d.	1	ENTER	
	1	ENTER	
	1	ENTER	ROOTS = –0.5 ± 0.866i

C. Polynomials

The formula defining the basic polynomial p of degree n or less is:

$$p(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \cdots + a_{n-1}x^{n-1} + a_nx^n$$

In the program the number of constants is N and the subcripts for the powers is from 0 to N–1. Let B(N) represent the value of each constant. The program is divided into two parts. The first **FOR. . . NEXT** loop places the constants into memory. The second part evaluates the polynomial and permits the user to reevaluate for different values of x.

Program:

```
10 "P": CLEAR: PAUSE "POLYNOMIALS"
20 INPUT "N="; N: DIM B(N)
30 FOR I=0 TO N–1
40 INPUT B(I)
```

```
50  NEXT I
60  INPUT "X="; X: P=0
70  FOR I=0 TO N-1
80  P=B(I)*X^I + P
90  NEXT I
100  PRINT "P=";P
110  INPUT "TRY NEW X(Y/N)?"; Q$
120  IF Q$="Y" THEN 60
130  END
```

Calculate p for the following equation for values of x from 1 to 10:

$$p(x) = 2x^4 - 8x^3 + 4x^2 - 2x + 1$$

Change to RUN Mode.

Key in:	DEF P		
N =	5	ENTER	
	1	ENTER	
	–2	ENTER	
	4	ENTER	
	–8	ENTER	*Answer:*
X =	1	ENTER	–3

X	2	3	4	5	6	7	8	9	10
P	–19	–23	57	341	997	2241	4337	7597	12381

D. Series

The values of many functions are approximations made with series. A series has an infinite number of terms. It is impossible to calculate every term in a particular series. The approximation improves with the number of terms included in the calculation.

For different values of x find the value of the definite integral:

$$\int_0^1 \mathrm{Sin}\, y^2 dy = \frac{x^3}{3} - \frac{x^7}{7(3!)} + \frac{x^{11}}{11(5!)} - \frac{x^{15}}{15(7!)}$$

Program:

```
10  “S”: CLEAR: PAUSE “SIN–Y–SQU”: N=3:G=1:L=0
20  INPUT “X=”; X
30  INPUT “NO. OF TERMS”; Y
35  REM ** VALUE OF EACH TERM **
40  P= X^N / (N * FACT G)
50  N=N+4: G=G+2: L=L+1
60  REM**MAKE EVEN TERMS NEGATIVE **
70  IF L/2–INT (L/2)=0 THEN LET P= –P
80  S=S+P
90  IF L<=Y THEN 40
100  PAUSE “SIN–Y–SQU=”
110  PRINT S
120  REM ** TO CHANGE # OF TERMS **
130  INPUT “TERMS #(Y/N)?”; Q$
140  IF Q$=“Y” THEN 30
150  END
```

Change to RUN Mode.

Key in: **DEF S** Number of terms and values of x

Answer:

	Value of X			
# of terms	1	.567	.1	.01
1	3.095238095E–01	6.031284904E–02	3.333309523E–04	3.333333331E–07
2	3.102813853E–01	6.03143242 E–02	Same	Same
4	3.102683028E–01	6.031432154E–02		
8	3.102683017E–01	Same		

LA definite integral commonly used in statistics is:

$$\int_0^x e^{-y^2}\,dy = x - \frac{x^3}{3} + \frac{x^5}{5(2!)} - \frac{x^7}{7(3!)}$$

Program:

```
10  "E": CLEAR: PAUSE "E-Y-SQU": N=1
20  INPUT "X=";X
30  INPUT "NO. OF TERMS";Y
35  REM ** VALUE OF EACH TERM **
40  P=X^N/(N*FACT L)
50  N=N+2: L=L+1
60  REM **MAKE EVEN TERMS NEGATIVE**
70  IF L/2-INT (L/2)=0 THEN LET P= -P
80  D=D+P
90  IF L<=Y THEN 40
100  PAUSE "E-Y-SQU="
110  PRINT D
120  REM **TO CHANGE # OF TERMS**
130  INPUT "TERMS #(Y/N)?";Q$
140  IF Q$="Y" THEN 30
150  END
```

Change to RUN Mode.

Key in:	DEF E	Terms:	*Answer*
X = 1		3	7.428571429E–01
		5	7.467291967E–01
		7	7.468228068E–01
		9	7.468241207E–01
		11	7.468241327E–01
X = 2		11	8.799081128E–01
		13	8.819118844E–01
		15	8.820712251E–01
		17	8.820809061E–01

F. Simpson's Rule

Simpson's Rule is a commonly used method for obtaining an approximation of the value of a definite integral. The area under a curve is divided into an even number of parts. The width of each part is given by:

$$\Delta X = \frac{b-a}{n}$$

and the area covered by the curve of the integral is:

$$\int_a^b f(x)dx = \frac{\Delta x}{3}(f_0 + 4f_1 + 2f_2 + 4f_3 + 2f_4 + \cdots + 4f_n + f_{n+1})$$

f_0 is $f(x)$ with $x = a$
f_1 is $f(x)$ with $x = a + \Delta x$
f_2 is $f(x)$ with $x = a + 2\Delta x$

$\ddots$

f_{n+1} is $f(x)$ with $x = b$

The integral to be evaluated is placed on Line 60 within the parenthesis. To obtain the area under the curve only three numbers need to be entered: the starting point on the x-axis (A), the ending point (B), and the number of parts (N) the area will be divided into.

Calculate the total value of the definite integral:

$$f(x) = x^3 + 2x^2 - x + 2$$

Program:

```
10   CLEAR: PAUSE "SIMPSONS RULE"
20   INPUT "B="; B, "A="; A, "N="; N
30   D=(B−A)/N
40   FOR I = 0 TO N
50   X = A + I * D
```

```
60   F = X^3 + 2 * SQU X-X + 2
70   T = L/2 - INT (L/2)
80   IF T>0 AND L<N THEN LET P = 4 * F
90   IF T=0 AND L<N-1 THEN LET P=2 * F
100  IF L=0 OR L=N THEN LET P=F
110  L=L+1: R=R +P
120  NEXT I
130  PAUSE "AREA ="
140  PRINT D * R/3
150  END
```

Change to RUN Mode.

Key in: B = 5 A = 1 N = 8

Answer: AREA = 234.66667

F. Matrices

A matrix is a rectangular array of numbers. The horizontal lists are called rows and the verticals, columns. A particular element in a matrix is identified by a row and column numbers:

A (row #, column #)

The following table is converted to a 3 * 3 matrix:

INVENTORY AIRPLANE PARTS

	BRASS	STEEL	COPPER
small	27	32	48
medium	82	51	77
large	15	28	19

MATRIX

$$A = \begin{vmatrix} 27 & 32 & 48 \\ 82 & 51 & 77 \\ 15 & 28 & 19 \end{vmatrix}$$

The first row and first column are denoted as zero. Therefore the number 27 is A(0, 0) and 51 is A(1, 1). I is the usual designation for the row rubscript and J for columns. The general term is:

A (I, J)

A matrix in BASIC is specified by a **DIM** statement.

Program:

```
10    DIM D(2, 2)
20    FOR I = 0 TO 2
30    FOR J = 0 TO 2
40    READ D(I, J)
50    NEXT J: NEXT I
60    DATA 27, 32, 48
70    DATA 82, 51, 77
80    DATA 15, 28, 19
```

To generalize user R and C as subscripts. Because 0 is used on the EL-5510, R and C will be one more than the actual number. To compensate, one is usually deducted in the **FOR** statements.

Program:

```
10    DIM D(R, C)
20    INPUT R, C
30    FOR I = 0 TO R–1
40    FOR J = 0 TO C–1
50    READ D(I, J)
```

```
60    NEXT J: NEXT I
70    DATA 27, 32, 48
80    DATA 82, 51, 77
90    DATA 15, 28, 19
```

Matrices may be added by adding corresponding elements assuming the matrices have the same number of elements and equal numbers of rows and columns.

An additional inventory of airplane parts is received with the following matrix:

$$B = \begin{vmatrix} 30 & 50 & 10 \\ 40 & 80 & 20 \\ 20 & 7 & 30 \end{vmatrix}$$

```
10    CLEAR: INPUT R, C
20    DIM D(R, C), B(R, C)
30    FOR I = 0 TO R–1
40    FOR J= 0 TO C–1
50    READ D(I, J)
60    NEXT J: NEXT I
70    FOR I = 0 TO R–1
80    FOR J = 0 TO C–1
90    READ B(I, J)
100   PRINT D(I, J) + B(I, J)
110   NEXT J: NEXT I
120   GOTO 30
130   DATA 27, 32, 48, 82, 51, 77, 15, 28, 19
140   DATA 30, 50, 10, 40, 80, 20, 20, 7, 30
150   END
```

Change to RUN Mode

Key in: RUN

```
? 3 ENTER
? 3 ENTER
```

Answer:

$$C = \begin{vmatrix} 57 & 82 & 58 \\ 122 & 131 & 97 \\ 35 & 35 & 49 \end{vmatrix}$$

A matrix may be decreased by subtraction in the same manner as addition or the **DIM** B can be multiplied by –1. The latter case is called scalar multiplication where a matrix is multiplied by a real number called a scalar.

To multiply the second inventory by 2 in the above program add the following line:

```
105 PRINT 2 * B(I, J)
```

Matrix multiplication can only be carried out if the number of columns in the first matrix equals the number of rows in the second matrix. If D(R, C) is to be multiplied by B(W, L); C must be equal to W. The resulting product will have the dimensions R * L; P(R, L).

Each value in the product is based on the equations:

$$\begin{array}{l} P(0,0) = D(0,0) * B(0,0) + D(0,1) * B(1,0) + \cdots + D(0,C) * B(C,0) \\ \quad \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ P(R,L) = D(R,0) * B(0,L) + D(R,1) * B(1,L) + \cdots + D(R,C) * B(C,L) \end{array}$$

Program:

```
10   CLEAR: INPUT "R=" ;R, "C=";C, "W =";W, "L=";L
20   DIM D(R, C), B(W, L), P(R, L)
30   REM **CHECK C=W**
40   IF C <> W THEN 10
50   FOR I=0 TO R–1
60   FOR J=0 TO C–1
70   INPUT "DATA1="; D(I, J)
80   NEXT J: NEXT I
```

```
90   FOR I=0 TO W–1
100  FOR J=0 TO L–1
110  INPUT "DATA 2="; B(I,J)
120  NEXT J: NEXT I
130  REM **MULTIPLICATION**
140  FOR I=0 TO R–1
150  FOR J=0 TO L–1
160  REM * CLEAR P–SUMS TO ZERO FOR NEXT PROGRAM LOOP
170  P(I,J)=0
180  REM * THIRD INDEX K NEEDED FOR MATCHING SUBSCRIPT
     IN EACH TERM
190  FOR K=0 TO C–1
200  P(I,J) = P(I,J) + D(I,K) * B(K,J)
210  NEXT K
220  PRINT P(I,J)
230  NEXT J: NEXT I
240  END
```

Production of three fruit juice drinks requires the following ounces/print of real juice:

TYPE OF DRINK

	A	B	C	D
ORANGE	1	2	4	4
LIME	3	2	3	1
LEMON	2	1	2	1

Production anticipated for the next 3 weeks.

		WEEK		
		1	2	3
DRINK:	A	45000	60000	50000
(pints)	B	49000	22000	70000
	C	42000	46000	57000
	D	19000	20000	21000

Calculate the required ounces of each fruit juice per week.

Change to RUN Mode

Key in: **RUN**

R = 3 **ENTER** W = 4 **ENTER**
C = 4 **ENTER** L = 3 **ENTER**

DATA 1 = 1 2 4 4 3 2 3 1 2
1 2 1

DATA 2 = 45000 60000 50000 49000 22000
70000 42000 46000 57000 19000
20000 21000

Answer: Requirements (Ounces)

WEEK	ORANGE	LIME	LEMON
1	387000	368000	502000
2	378000	382000	482000
3	242000	254000	305000

In many applications of matrices the determinant of a square matrix must be calculated. The determinant of a 3 * 3 matrix is given as:

$$\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = a_1 b_2 c_2 + a_2 b_3 c_1 + a_3 b_1 c_2 - a_1 b_3 c_2 - a_3 b_2 c_1 - a_2 b_1 c_3$$

Program:

```
10    CLEAR: DIM B(2, 2)
20    FOR I = 0 TO 2
30    FOR J = 0 TO 2
40    INPUT "DATA ="; B(I, J)
50    NEXT J: NEXT I
60    G = B(0, 0) * B(1, 1) * B(2, 2) + B(1, 0) * B(2, 1) * B(0, 2) +
      B(2, 0) * B(0, 1) * B(1, 2)
70    F = -B(0, 0) * B(2, 1) * B(1, 2) - B(2, 0) * B(1, 1) * B(0, 2) -
      B(1, 0) * B(0, 1) * B(2, 2)
80    PRINT "D="; G + F
90    END
```

Calculate the Determinant for the matrix:

$$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix}$$

Change to RUN Mode

Key in: RUN

Data = 1 2 3 4 5 6 7 8 9

Answer: D = 0

Change the number 9 to 10

Answer: D = –3

APPENDIX

Glosary of Terms

annunciators Small messages that are in the screen of the EL-5510 that the computer uses to communicate what mode it is in. **BUSY**, **DEG**, **RAD**, **GRAD**, **P**, and **E** are all used by the EL-5510.

array A set of variables denoted by subscripting of a single variable name. Can be either numeric or string.

ASCII The set of codes used in many computers to represent characters.

BASIC A language for communicating with a computer. The letters stand for Beginners All-purpose Symbolic Instruction Code. One of the most widely used languages in existence today.

binary A system of numbers comprised of two digits: 0 and 1. See decimal, hexadecimal, and octal.

binary number A number expressed in Binary.

bit A binary digit; either a 0 or a 1.

bug A problem or error in a program that causes the program to not run properly. Almost always due to insufficient planning of a program or a typographical error in entering a program into the computer's memory.

byte A binary number that is eight binary bits or three octal digits or two hexadecimal digits long. Generally can be thought of as one character.

character Any piece of information that can be stored in one byte. Examples are the letters of the alphabet.

character codes The single byte numeric codes that identify each ASCII code. These codes may differ from one computer to onother. See ASCII.

code The instructions used to write a program.

concatenation The combining of two or more strings of characters to form one larger string.

data A general term used to describe both alphabetic (string) information and numeric information.

debugging The art of finding and fixing bugs in a program.

decimal The number system that humans use. It has ten individual digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

digit A single numeric character.

edit To change a section of program code.

enter The command that generally tells the computer that you are finished entering information into it through the key-board. This command is executed on the EL-5510 by pressing the **ENTER** key.

EPROM A type of memory used to store information in that doesn't need constant electrical power to maintain. The letters stand for Eraseable Programmable Read Only Memory.

firmware A term used to describe operating instructions for the computer that are generally found in ROM.

flowchart A method of outlining the logic of a program using diagrams.

hardware Computer equipment as opposed to computer programs. See software.

hexadecimal Also called **hex** for short. A number system based on sixteen different digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Each hex digit represents for binary bits.

integer A positive or negative number which has no fractional part.

integrated circuit Also known as a "chip" or an "IC." It is a type of circuit that is very, very small. IC's are used to replace larger and generally more expensive circuit components.

kilobyte (K) A unit of computer memory which represents 1,024 bytes. Casually used to describe 1000 byte chunks of memory.

language A set of commands and statements that allow humans to communicate with computers. The EL-5510 uses BASIC.

laser Light Amplification through Stimulated Emission of Radiation. Lasers are a type of very concentrated and controllable light sources that can be used to do everything from welding a car to micro-surgery.

literal A value, either numeric or string, that is part of a program's code. As opposed to a variable.

loop A section of program code that does the same thing over and over.

machine code A method of programming a computer using binary, hexadecimal, octal, or decimal numbers to represent instructions. Very tedious but very powerful.

mass storage device This phrase is used to describe any number of pieces of machinery that can be used by a computer to store extremely large quantities of information. Some of the most common types are magnetic tapes and disks.

memory Electronic circuits in a computer that can be used to store information. Memory can be in many different forms: RAM, ROM, PROM, and EPROM.

nesting The practice of having one loop inside another loop.

number A character or set of characters that can be used as arithmetic operands in an equation or calculation.

octal A number system based on eight individual digits: 0, 1, 2, 3, 4, 5, 6, 7. Used instead of hex by some computer system to represent groups of binary bits. Each octal digit represents three binary bits.

program The set of instruction written by the programmer that teaches the computer what to do.

PROM Programmable Read Only Memory.

RAM The section of computer memory that is used to write programs and to store data in the form of variables. The letters stand for Random Access Memory.

ROM The section of memory used by the computer to contain the BASIC interpreter language instructions. It is neither changeable nor usable by the programmer. The letters stand for Read Only Memory.

routine A program or set of instructions for a computer to execute.

scientific notation A method of representing decimal numbers in powers of ten. It enables the computer to store very large numbers in a very small amount of memory.

software Computer programs.

statement An instruction ot the computer in a program.

string Characters that cannot be used as numeric operands. See **number**.

subroutine A section of a program that is referenced or called by another section of the same program.

syntax The proper way in which to structure an instruction to the computer so that the BASIC interpreter can understand the instruction and carry it out.

INDEX

A

- **ACC** 13
- Accelerated cost recovery system 69
- Acid-Test 6
- Addition 5
- Algorithm 34
- Alphanumeric 39
- Amortizing 59
- **AMRT** 13
- **AND** 54
- Annual percentage rate 15
- **APO** 6
- **APR** 13
- **AREAD** 51
- Array 52
- **ASC** 55
- ASCII code 55
- Automatic power off 6
- Average, mean 26
  - moving 73

B

- Balance sheet 24
- Balloon payment 24
- BASIC 36
- **BEEP** 42
- Begin 14
- **BGN** 14
- **BGNON** 67
- Bonds 18
- Break-even 62
- **BRK** 39

C

- Calculator mode 2
- Calendar 11
- Call date 18
- Cash, discounted 25
  - flow 25
- **CD** 30
- CFi 14
- Chi-square test 84
- **CHR$** 55
- **CLEAR** 48
- **CLOAD** 58
- Code 38
- Comma 39
- **COMP** 11
- Complex number 88
- Concatenation 52
- Conditionals 43
- Constants 7

CONT 39
Conversion, feet/inches. 86
Correct data 30
Correlation. 30
Costs, fixed 62
    variable. 62
Coupons 18
CSAVE 58
CST 11
Cursor. 41

D

Data 26
DATA. 49
Dates, days between. 11
Debugging 58
Debt 18
DEF 51
Definite integral 91
Degrees of freedom 83
Delete 41
Delta percent 9
Demand. 62
Depletion. 17
Depreciation. 68
Determinant. 99
Deviance 27
DIM 52
Discounted cash flow 25
Discounts, loan 23
    trade 8
Discriminant. 87
Distribution, chi-square 84
    normal 27
    "t" 82
Dividends. 27
Division. 6
Documentation. 34

E

Editing 41
EFF 15
Effective rate of interest 15
Elements 96
END 38
ENTER 38
Equillibrium. 65
ERASE FIN 48
Exponential 32
Exponential regression 32
Extrapolation 32

F

Face value 18
Factorial 91
Fees 23
FIN mode 2
Flag 50
Flowcharting 35
FOR . . . TO. 45
Frequency, expected 84
Future value. 22

FV 13

G

GOSUB 48
GOTO 42

I

i 14
IF . . . THEN 42
Index numbers 8
INKEY$ 56
INPUT 42
INPUT# 59
Insert 41
INT 70
Integral 91
Interest, compounded 14
  paid 21
  rate. 14
Internal rate of return 14
Inventory. 9
IRA 79
IRR 14

L

Lease 65
LEFT$ 54
LEN 54
LET 45
Line of code. 38
Linear regression 30
LIST. 41
Loans 40
Log, natural 53
Loop. 45
LPRINT 59

M

Makeham's formula 19
MAR. 11
Markup 10
Matrices. 94
Matrix, multiplication 97
Maturity 18
MDF. 47
Mean. 27
Memory, recall 9
  store 9
MID$ 54
Mode, BASIC 1, 36
  CAL 1
  FIN. 2
  Normal. 2
  PRO 2
  RUN 2
  STAT. 2
Mortgages 40
Moving averages 73
MU. 11
Multiplicand. 7
Multiplication, matrix 97

- Multiplication, use of 6

N

- **n** 12
- Natural log. 33
- Nested looping 46
- Net present value. 25
- **NEXT** 45
- **NEW**. 38
- Ni. 14
- Normal distribution 27
- **NOT** 54
- **NPV** 14

O

- **ON**. 51
- **OR** 54

P

- **P1/P2** 13
- Parenthesis. 93
- **PASS** 57
- **PAUSE** 39
- Percent 9
- Percent change 9
- **PMT** 13
- Points 22
- Polynomial. 89
- Population 28
- Power 6
- Present value 22
- Price 62
- Price/earnings 7
- Principal paid 21
- **PRINT** 39
- **PRINT#** 59
- Programming 34
- Purchase vs. lease. 65
- **PV** 13

Q

- Quadratic, equations 87
  - roots 87
- Quotation marks 87

R

- **RANDOM** 57
- Random numbers 57
- Ratios 2
- **RCL** 22
- **READ**. 49
- Rebate 77
- Regression, exponential 32
  - linear 30
- **REM**. 48
- **RESTORE** 49
- **RETURN**. 48
- **RIGHT$** 54
- **RND**. 57
- Roots, imaginary. 88
  - quadratic 87
  - real 88

Rule of 78 77
RUN 39

S

Sample, size 29
Savings 16
SEL 11
Semicolon 39
Serial Bond 19
Series 90
Sigma 26
Simpson's rule 93
INT 21
Sinking fund. 20
Slope 30
PRN 21
Standard deviation. 27
STAT mode 28
Statistics 82
STEP 45
STO 26
STOP 39
Store memory. 10
STR$ 59
Strings. 39
Subroutine. 48
Subscript 52
Subtraction 6
Supply 62

T

"t" test 82
TAB 6
Tape storage. 58
TROFF 58
TRON. 58
trust fund 16

U

USING 47

V

VAL. 56
Value, future 22
Variance 28

W

WAIT 39
Weighting. 28
Wholesale. 10

SHARP

DOLLARS AND CENTS COMPUTING

In today's economic climate, it's more important than ever to make the right financial choices. More options are available, more factors to be considered. And in such a climate, the difference between success and failure usually depends on being able to evaluate and select the right options.

The EL-5510 can help. It combines a powerful financial calculator with a BASIC-language computer. Many problems can be solved directly in the calculator mode. Complex or frequently-needed problems can be programmed in BASIC for quick solution anytime. The 5510's built-in financial functions can solve problems from simple profit margin to compound interest, including:

- Savings
- Loans
- Mortgages
- Leases
- Real Estate
- IRA problems
- Depreciation
- Discounted Cash Flow Analysis
- Amortization of payment schedules
- Stocks and bonds
- Inflation
- Statistics
- Balloon payments
- Break-even analysis

This text provides exact key sequences for solution of many of the above problems. With this text as a guide, the EL-5510 can take on the financial challenges of the 80's.

Printed in Japan
4M3.7T(TMANE1074CCZZ)②